CA-IDMS®

Installation and Maintenance — z/OS and OS/390

15.0

Computer Associates™

This documentation and related computer software program (hereinafter referred to as the “Documentation”) is for the end user's informational purposes only and is subject to change or withdrawal by Computer Associates International, Inc. (“CA”) at any time.

This documentation may not be copied, transferred, reproduced, disclosed or duplicated, in whole or in part, without the prior written consent of CA. This documentation is proprietary information of CA and protected by the copyright laws of the United States and international treaties.

Notwithstanding the foregoing, licensed users may print a reasonable number of copies of this documentation for their own internal use, provided that all CA copyright notices and legends are affixed to each reproduced copy. Only authorized employees, consultants, or agents of the user who are bound by the confidentiality provisions of the license for the software are permitted to have access to such copies.

This right to print copies is limited to the period during which the license for the product remains in full force and effect. Should the license terminate for any reason, it shall be the user's responsibility to return to CA the reproduced copies or to certify to CA that same have been destroyed.

To the extent permitted by applicable law, CA provides this documentation “as is” without warranty of any kind, including without limitation, any implied warranties of merchantability, fitness for a particular purpose or noninfringement. In no event will CA be liable to the end user or any third party for any loss or damage, direct or indirect, from the use of this documentation, including without limitation, lost profits, business interruption, goodwill, or lost data, even if CA is expressly advised of such loss or damage.

The use of any product referenced in this documentation and this documentation is governed by the end user's applicable license agreement.

The manufacturer of this documentation is Computer Associates International, Inc.

Provided with “Restricted Rights” as set forth in 48 C.F.R. Section 12.212, 48 C.F.R. Sections 52.227-19(c)(1) and (2) or DFARS Section 252.227-7013(c)(1)(ii) or applicable successor provisions.

Second Edition, October 2001

© 2001 Computer Associates International, Inc.
All rights reserved.

All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.

Contents

How to use this manual xiii

Chapter 1. Introduction 1-1
1.1 New Install and Installation Guide Features 1-3
1.2 Combined Installation Manual 1-4
Example 1-5
1.3 CA-IDMS Installation Types 1-7
1.3.1 Complete Base Installation 1-7
1.3.2 Upgrade Installation 1-8
1.3.3 Add-On Installation 1-8
1.4 Using the New CICS Interface 1-10
1.4.1 Streamlined CICS Interface Installation 1-10

Chapter 2. Installation Prerequisites 2-1
2.1 Hardware Requirements 2-4
2.1.1 Mainframe 2-4
2.1.1.1 CPU Requirements 2-4
2.1.2 Disk Drives 2-4
2.1.2.1 Disk Space 2-4
2.1.3 Tape Drive 2-5
2.2 Software\Interface Requirements 2-6
2.2.1 Operating System 2-6
2.2.2 SVC 2-6
2.2.3 Unicenter TNG Framework for OS/390 2-6
2.2.3.1 CA LMP Keys 2-7
2.2.4 VTAM 2-7
2.2.5 CICS 2-7
2.2.6 Libraries 2-8
2.2.7 SMP/E 2-8

Chapter 3. Getting Started 3-1
3.1 Overview of Install Steps 3-4
3.2 Review Cover Letters and PMLs 3-5
3.3 CA LMP Keys 3-6
Example 3-8
3.4 Remove USERMODs 3-9
Example 3-10

Chapter 4. Installation Process 4-1
4.1 Overview 4-3
4.2 Download SAMPJCL 4-4
4.3 Download CAISAG Program and Skeleton 4-5
4.4 Customize the CAISAG Input 4-6
4.4.1 JOBCARD 4-6
4.4.2 PASSWORD 4-6
4.4.3 SETUP 4-6
4.4.4 SETUPBK 4-7

4.4.5 VARBLIST 4-7
4.5 Run CAISAG 4-9

Chapter 5. VARBLIST Variables 5-1
5.1 Common Variables 5-4
5.1.1 Member Name 5-4
Example 5-4
5.1.2 Products 5-4
5.1.3 Global DMCL 5-4
5.2 CA-IDMS Variables 5-5
5.2.1 Upgrade 5-5
5.2.2 Tape Class 5-5
5.2.3 Disk VOLSER and Disk Contention 5-5
5.2.4 SMP/E Environment 5-6
5.2.5 CV Number 5-7
5.2.6 DC System Number 5-7
5.2.7 Free Storage 5-7
5.2.8 Access Method 5-8
5.2.9 Uppercase Terminal Support 5-8
5.2.10 Authorized Userid 5-8
5.2.11 Link WTOEXIT with Startup Module 5-8
5.2.12 Startup Module 5-8
5.2.13 Dynamic or Static PDE Support 5-9
Example 5-9
5.2.14 Storage Protection 5-9
5.2.15 Backup SMP/E Environment 5-10
5.2.16 VTAM ID 5-10
5.2.17 CWA Displacement 5-11
5.2.18 CA-Culprit 5-11
5.2.19 Add-On Installation Variables 5-11
5.2.20 Page Size 5-11
5.2.21 Unicenter TNG Framework for OS/390 5-12
5.3 CA-IDMS Tools Variables 5-13
5.3.1 Upgrade 5-13
5.3.2 Create a CA-IDMS Tools Dictionary 5-13
5.4 CA-Endevor/DB Variables 5-14
5.4.1 Existing CA-Endevor/DB into a New SMP/E Environment 5-14
5.4.2 Upgrade CA-Endevor/DB 5-14
5.4.3 CA-Endevor/DB Message Library 5-14

Chapter 6. CA-IDMS Installation Jobs 6-1
6.1 Create SMP/E Environment 6-4
6.1.1.1 Restart Information 6-4
6.2 Download Modules 6-5
6.2.1.1 CAIIPDS Return Codes 6-5
6.2.1.2 Restart Information 6-5
6.3 Create Customized Source 6-6
6.3.1 IDMSOPTI 6-6
Example 6-6
6.3.1.1 Restart Information 6-7
6.4 SMP/E RECEIVE Processing 6-8

6.4.1.1 Restart Information 6-8
Example 6-8
6.5 SMP/E APPLY Processing 6-9
6.5.1.1 Restart Information 6-9
6.6 SMP/E ACCEPT Processing 6-10
6.6.1.1 Restart Information 6-10
6.7 Miscellaneous SMP/E Processing 6-11
6.7.1.1 Restart Information 6-11
6.8 Backup SMP/E Environment 6-12
6.8.1.1 Restart Information 6-12
6.9 Create USERMOD for WTOEXIT 6-13
6.9.1.1 Restart Information 6-13
6.10 Load the SVC Using CAIRIM 6-14
6.10.1.1 Return Codes 6-15
6.10.1.2 Restart Information 6-15
6.11 Allocate Database Files 6-16
6.11.1.1 Restart Information 6-16
6.12 Create SYSTEM and SYSDIRL Dictionaries 6-17
6.12.1.1 Return Codes: 6-17
Example 6-17
6.12.1.2 Restart Information 6-17
6.13 Create APPLDICT and ASF Dictionaries 6-18
6.13.1.1 Restart Information 6-18
6.14 Build Non-SQL Commonweather Database 6-19
6.14.1.1 Restart Information 6-19
6.15 Create SQL Demo Database 6-20
6.15.1.1 Restart Information 6-20
6.16 Generate SYSTEM90 and Format Journals 6-21
6.16.1.1 Restart Information 6-21
6.17 Create UCFCICS Load Module 6-22
6.17.1.1 Restart Information 6-22
6.18 Backup Database Files 6-23
6.18.1.1 Restart Information 6-23
6.19 Populate SAMPJCL 6-24
6.19.1.1 Restart Information 6-24

Chapter 7. CA-IDMS Post-Installation Tasks 7-1
7.1 Environment Created by Installation 7-4
7.2 OS/390 Environment 7-5
7.3 Prepare TP Access Environment 7-6
7.3.1 VTAM Access 7-6
7.3.2 TSO Access 7-6
7.3.3 CICS Access 7-7
7.3.3.1 CICS Access to Multiple DC/UCF Systems 7-7
Program Interface Module 7-7
Interface Modules 7-8
7.3.3.2 Multiple CICS Regions 7-8
System Table for the Back-end 7-8
TPNAME 7-9
Front-end Table 7-9

Example . . . 7-10
Front-end Table Load Module . . . 7-10
CICS PPT . . . 7-10
7.4 Creating an Executable System . . . 7-11
Copying System 90 . . . 7-11
Defining a New System . . . 7-11
7.5 Verify the System is Installed . . . 7-12
7.5.1 System Startup . . . 7-12
7.5.2 Online Verification . . . 7-12
7.5.3 Batch Verification . . . 7-13
7.6 CAIRIM . . . 7-15
7.7 SVC . . . 7-16
Example . . . 7-16
Explanation. . . . 7-16
7.8 Optional APARs . . . 7-17
7.9 USERMODs . . . 7-18
7.10 Conversion of a 10.2 System . . . 7-19
7.11 Security . . . 7-20

Chapter 8. CA-IDMS Tools Installation Jobs . . . 8-1
8.1 Update the CA-IDMS SMP/E Environment . . . 8-4
8.1.1.1 Restart Information . . . 8-4
8.2 Download Modules from Tape . . . 8-5
8.2.1.1 CAIIPDS Return Codes . . . 8-5
8.2.1.2 Restart Information . . . 8-5
8.3 Customize CA-IDMS Tools Runtime Options . . . 8-6
8.3.1.1 Restart Information . . . 8-6
8.3.2 CA-IDMS/DML Online Customization . . . 8-7
8.3.3 CA-IDMS/DC SORT Considerations . . . 8-7
8.3.3.1 Examples of CA-IDMS/DC SORT Customization . . . 8-8
Example 1 . . . 8-8
Example 2 . . . 8-8
Example 3 . . . 8-10
Example 4 . . . 8-10
8.4 SVC for CA-IDMS Tools . . . 8-11
Batch Jobs . . . 8-11
CICS Tasks . . . 8-11
8.4.1.1 Restart Information . . . 8-11
8.5 SMP/E RECEIVE . . . 8-12
8.5.1.1 Restart Information . . . 8-12
Example . . . 8-12
8.6 SMP/E APPLY Processing . . . 8-13
8.6.1.1 Restart Information . . . 8-13
8.7 SMP/E ACCEPT Processing . . . 8-14
8.7.1.1 Restart Information . . . 8-14
8.8 Miscellaneous SMP/E Processing . . . 8-15
8.8.1.1 Restart Information . . . 8-15
8.9 Product-Specific Database Installation Tasks . . . 8-16
8.9.1.1 Return Codes . . . 8-16
8.9.1.2 Restart Information . . . 8-16
8.10 CA-IDMS Tools User Exits . . . 8-17

8.10.1.1 Restart Information 8-17
8.10.2 CA-IDMS Tools User Exits 8-17
8.10.2.1 Determining Required SAMPJCL Members 8-20
Example 8-21

Chapter 9. CA-IDMS Tools Post-Installation Tasks 9-1
9.1 Install a Special SVC (SMF Use Only) 9-4
9.2 Convert the CA-IDMS/Database Extractor Database 9-5
9.3 Update the Dictionary 9-6
9.3.1 CA-IDMS/ADS Trace Dictionary Updates 9-6
9.3.2 CA-IDMS/DC SORT Dictionary Updates 9-6
9.3.3 CA-IDMS/DMLO Dictionary Updates 9-6
9.3.4 CA-IDMS/DQF Dictionary Updates 9-7
9.4 Update the CICS Tables 9-8
9.4.1 CA-IDMS/DC SORT 9-8
9.4.2 CA-IDMS/DMLO 9-8
9.5 CA-IDMS/DMLO for TSO 9-9
9.6 Modifying the Sysgen 9-10
9.7 Modify CA-IDMS Startup JCL 9-12
9.8 Cycle CA-IDMS System 9-13
9.9 Install Default JCL 9-14

Chapter 10. CA-Endevor/DB Installation Jobs 10-1
10.1 Update the CA-IDMS SMP/E Environment 10-4
10.1.1.1 Restart Information 10-4
10.2 Download Modules from Tape 10-5
10.2.1.1 CAIIPDS Return Codes 10-5
10.2.1.2 Restart Information 10-5
10.3 SMP/E RECEIVE Processing 10-6
10.3.1.1 Restart Information 10-6
Example 10-6
10.4 SMP/E APPLY Processing 10-7
10.4.1.1 Restart Information 10-7
10.5 SMP/E ACCEPT Processing 10-8
10.5.1.1 Restart Information 10-8
10.6 Database Installation Tasks 10-9
10.6.1.1 Restart Information 10-9

Chapter 11. CA-Endevor/DB Post Installation Tasks 11-1
11.1 Modify CA-IDMS Startup JCL 11-3
11.2 Cycle CA-IDMS 11-4

Chapter 12. Maintenance Process 12-1
12.1 Test Fixes and APARs 12-4
12.1.1.1 Installing APARs or Test Fixes: 12-4
Error Messages 12-4
12.1.1.2 Removing APARS and Test Fixes 12-5
12.1.1.3 Test Fixes and Corresponding APARs 12-5
12.1.1.4 ACCEPTing Test Fixes, Optional APARs and Published APARS 12-6
12.1.2 APAR tapes 12-6

12.1.3 Informational APAR, or PML (Product Maintenance Letter) 12-6
12.2 Maintenance Tapes 12-7
12.3 Installing Maintenance Tapes 12-8
12.3.1 Review the Cover Letter 12-8
12.3.2 Download Maintenance SAMPJCL Library 12-8
12.3.3 Review the #README Member 12-9
12.3.4 Customize the JOBCARD Member 12-9
12.3.5 Customize the SMP/E Procedure 12-9
12.3.6 Remove USERMODS 12-9
Example 12-9
12.3.7 RECEIVE Maintenance 12-10
12.3.8 Run Genlevel-specific Jobs 12-10
12.3.9 APPLY CHECK Maintenance 12-10
12.3.10 RESTORE Applicable SYSMODs 12-10
12.3.11 APPLY Maintenance 12-10
12.3.11.1 Non-zero Condition Codes 12-11
12.3.12 ACCEPT CHECK Maintenance 12-11
12.3.13 ACCEPT Maintenance 12-11
12.3.13.1 Non-zero Condition Codes 12-11
12.3.14 Backup SMP/E 12-11
12.3.15 Reinstall User Modifications 12-12
12.3.16 Re-APPLY Applicable SYSMODs 12-12
12.3.16.1 Non-zero Condition Codes 12-12
12.3.17 Install Special Processing Maintenance 12-13
12.3.18 Save all Materials and Output 12-13

Chapter 13. User Modification Process 13-1
13.1 Overview of User Modifications 13-4
13.2 USERMOD Syntax 13-5
13.3 Required Statements 13-6
13.3.1 ++USERMOD 13-6
Example 13-6
13.3.2 ++VER 13-6
Example 13-6
13.4 Element Replacement 13-7
Example 13-7
13.5 Element Updates 13-8
Example 13-8
13.6 JCLIN 13-9
Example 13-9
Example 13-10
13.7 JCL to Install USERMODs 13-11
13.7.1 RECEIVE Step 13-11
13.7.2 APPLY Steps 13-11
13.7.2.1 Input Libraries 13-12
13.7.2.2 Output Libraries 13-12
Example 13-12
13.8 Installing USERMODs 13-13
13.8.1 MODID 13-13
13.9 REDO 13-14
13.10 Restoring USERMODs 13-15

13.11 SUP Parameter 13-16
13.12 MULTIPLE CVs 13-17
13.13 GIM65901W 13-18
13.14 Multiple Load Module Versions (GIM41101E) 13-19
Example 13-19

Appendix A. CA-IDMS Product List A-1
A.1 CA-IDMS Products A-4
A.1.1 CA-IDMS Base Products A-4
A.1.2 Tools Products in CA-IDMS Family A-4
A.1.3 CA-IDMS Transparency Products A-5
A.1.4 CA-IDMS TP Access Interfaces A-5
A.1.5 CA-IDMS/Culprit™ Interfaces A-5
A.2 CA-IDMS Tools Products A-7
A.3 CA-Endevor/DB A-8

Appendix B. Installation Tape Description B-1
B.1 Installation Tape Format B-3
B.2 Installation Tape Contents B-4

Appendix C. VARBLIST C-1
C.1 CA-IDMS VARBLIST Member Listing C-3
C.2 CA-IDMS Tools VARBLIST Member Listing C-23
C.3 CA-Endevor/DB VARBLIST Member Listing C-39

Appendix D. SMP/E D-1
D.1 Overview of SMP/E D-4
D.1.1 Non-SMP/E Install D-4
D.1.2 Target and Distribution Libraries D-4
D.1.3 SMP/E Invokes Utilities D-5
D.1.4 MCS D-5
D.1.5 CSI D-5
Example D-6
D.1.6 DDDEFs D-7
D.1.7 Indirect Libraries D-7
D.1.8 JCL D-7
D.1.9 RECEIVE/APPLY/ACCEPT D-8
D.1.10 FMID D-9
D.1.11 SYSMOD D-10
D.1.12 Sample SMP/E job D-10
D.1.13 CSI Entries D-11
Example D-11
D.1.14 CHECK Parameter D-12
D.1.15 Reversing Processing D-12
D.1.16 LMOD D-12
Example. D-13
D.1.17 SOURCEID D-13
Example D-13
D.1.18 Relfiles D-14
D.1.19 LIST Command D-14

Example D-14
Example D-15
D.1.20 Restarting an SMP/E Job D-15
D.1.21 In Conclusion D-15
D.2 Debugging SMP/E Jobs D-16
D.2.1 Order of Processing D-16
D.2.2 Output Datasets D-16
D.2.3 Error Message Flag D-17
D.2.4 Syntax/Conflict Checking D-17
D.2.5 Copy Failure D-17
D.2.6 Zap Failure D-17
Resolution. D-17
D.2.7 Assembly Failure D-18
D.2.8 Link Failure D-18
Example D-18
D.2.9 SMPLOG D-18
D.2.10 Recovery D-18
D.2.11 Support D-19
D.2.12 ASMA417C D-19
D.2.13 SYSLIB D-19
D.2.14 MODID Error D-20
D.2.15 MODID Warning D-21
D.2.16 Superceding (SUP) APARs and Test Fixes D-22
D.2.17 Removing APARs D-22

Appendix E. CICS Macros E-1
E.1 CICSOPT Macro E-4
CICSOPT syntax E-4
CICSOPT Parameters E-5
Assembling CICSOPTS and link editing IDMSINTC E-10
OS/390 CICSOPTS assembly and IDMSINTC link edit E-10
E.2 IDMSCINT E-12
IDMSCINT syntax E-12
IDMSCINT parameters E-12

Appendix F. CA-Culprit Profile Options F-1
F.1 Keywords and Operands F-4
F.1.1 Block/Track Option F-4
F.1.2 CALC Key Sign Option F-4
F.1.3 Headers Option F-4
F.1.4 Assembly Date Option F-4
F.1.5 Buffer Size Option F-5
F.1.6 Lines Per Page Option F-5
F.1.7 Date Stamp Option F-5
F.1.8 Error Options F-5
F.1.9 Report Error Level Option F-6
F.1.10 Hexadecimal Dump Option F-6
F.1.11 IDMS Buffer Size Option F-6
F.1.12 DDNAME Modification Option F-6
F.1.13 Report Lines Per Page Option F-7
F.1.14 Line Size Option F-7

F.1.15 Numeric Editing Option F-7
F.1.16 Operating System Option F-7
F.1.17 Tape Records/Block Option F-7
F.1.18 Return Codes Option F-8
F.1.19 Relocating Loader Option F-8
F.1.20 Schema Name Option F-8
F.1.21 Repeat First Page Option F-8
F.1.22 SPIE/STXIT Routine Option F-8
F.1.23 File Characteristics Option F-9
F.1.24 Time Stamp Option F-10
F.1.25 Separator Character Option F-10
F.1.26 Source Library Option F-10
F.1.27 CA-Panvalet File Option F-10

Appendix G. CA-IDMS Tools Runtime Options G-1
G.1 CA-IDMS/ADS Alive Runtime Parameters G-4
G.2 CA-IDMS/Database Extractor Runtime Parameters G-5
G.3 CA-IDMS/Dictionary Migrator Runtime Parameters G-7
G.4 CA-IDMS/Dictionary Migrator Assistant Runtime Parameters G-22
G.5 CA-IDMS/Dictionary Module Editor Runtime Parameters G-23
G.6 CA-IDMS/Dictionary Query Facility Runtime Parameters G-25
G.7 CA-IDMS/DML Online Runtime Parameters G-26
G.8 CA-IDMS/Enforcer Runtime Parameters G-37
G.9 CA-IDMS/Master Key Runtime Parameters G-38
G.10 CA-IDMS/Online Log Display Runtime Parameters G-39
G.11 CA-IDMS/SASO Runtime Parameters G-40
G.12 General Sort Runtime Parameters G-41

Appendix H. CA-IDMS/DMLO Implementations H-1
H.1 CA-IDMS/DMLO Security and Access H-4
H.1.1 CA-IDMS/DMLO Security H-4
H.1.2 CA-IDMS/DMLO Access Restrictions H-5
H.1.2.1 Restricting Usage Mode Access Globally H-5
Example H-5
H.1.2.2 Restricting Usage Mode Access by User H-5
Example H-5
H.1.2.3 Central CA-IDMS Security H-6
H.2 Implementing CA-IDMS/DMLO in Multiple CV's Under CICS H-7

Index X-1

How to use this manual

This installation guide contains information to install and maintain Computer Associates' CA-IDMS and related products on the OS/390 and z/OS operating sytems. The CA-IDMS 15.0 documentation set describes procedures and JCL for the OS/390 environment. The same procedures and JCL are applicable for the z/OS environment.

Step-by-step procedures to install, customize, and maintain the CA-IDMS product line are provided. See Appendix A, "CA-IDMS Product List" for a complete list of CA-IDMS products.

Note: CAIRIM and optionally other components of Unicenter TNG Framework for OS/390 must be installed prior to installing CA-IDMS. Refer to Chapter 2, "Installation Prerequisites" for specific information regarding Unicenter TNG Framework for OS/390 component requirements.

See the *Unicenter TNG Framework for OS/390 Getting Started Guide* for instructions on installing Unicenter TNG Framework for OS/390 services.

Organization

This document is divided into the following chapters:

Chapter	Description
1	Chapter 1, “Introduction”
2	Chapter 2, “Installation Prerequisites”
3	Chapter 3, “Getting Started”
4	Chapter 4, “Installation Process”
5	Chapter 5, “VARBLIST Variables”
6	Chapter 6, “CA-IDMS Installation Jobs”
7	Chapter 7, “CA-IDMS Post-Installation Tasks”
8	Chapter 8, “CA-IDMS Tools Installation Jobs”
9	Chapter 9, “CA-IDMS Tools Post-Installation Tasks”
10	Chapter 10, “CA-Endevor/DB Installation Jobs”
11	Chapter 11, “CA-Endevor/DB Post Installation Tasks”
12	Chapter 12, “Maintenance Process”
13	Chapter 13, “User Modification Process”
Appendix A	Appendix A, “CA-IDMS Product List”
Appendix B	Appendix B, “Installation Tape Description”
Appendix C	Appendix C, “VARBLIST”
Appendix D	Appendix D, “SMP/E”
Appendix E	Appendix E, “CICS Macros”
Appendix F	Appendix F, “CA-Culprit Profile Options”
Appendix G	Appendix G, “CA-IDMS Tools Runtime Options”
Appendix H	Appendix H, “CA-IDMS/DMLO Implementations”

Related Publications

You may need to reference the documents listed below during the installation process. To order additional documentation, you may visit our support website at *http://esupport.ca.com/* or call *1-800-637-5858*.

Unicenter TNG Framework for OS/390 documents

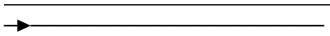
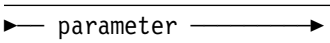
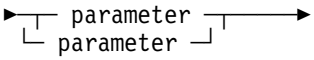
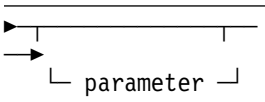
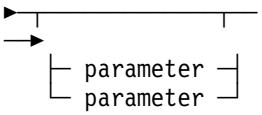
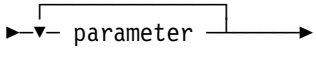
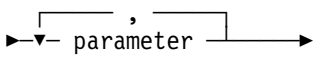
- *Unicenter TNG Framework for OS/390 Administrator Guide*
- *Unicenter TNG Framework for OS/390 Getting Started Guide*
- *CA Message Guide*
- *CA Reference Guide*

CA-IDMS Documents

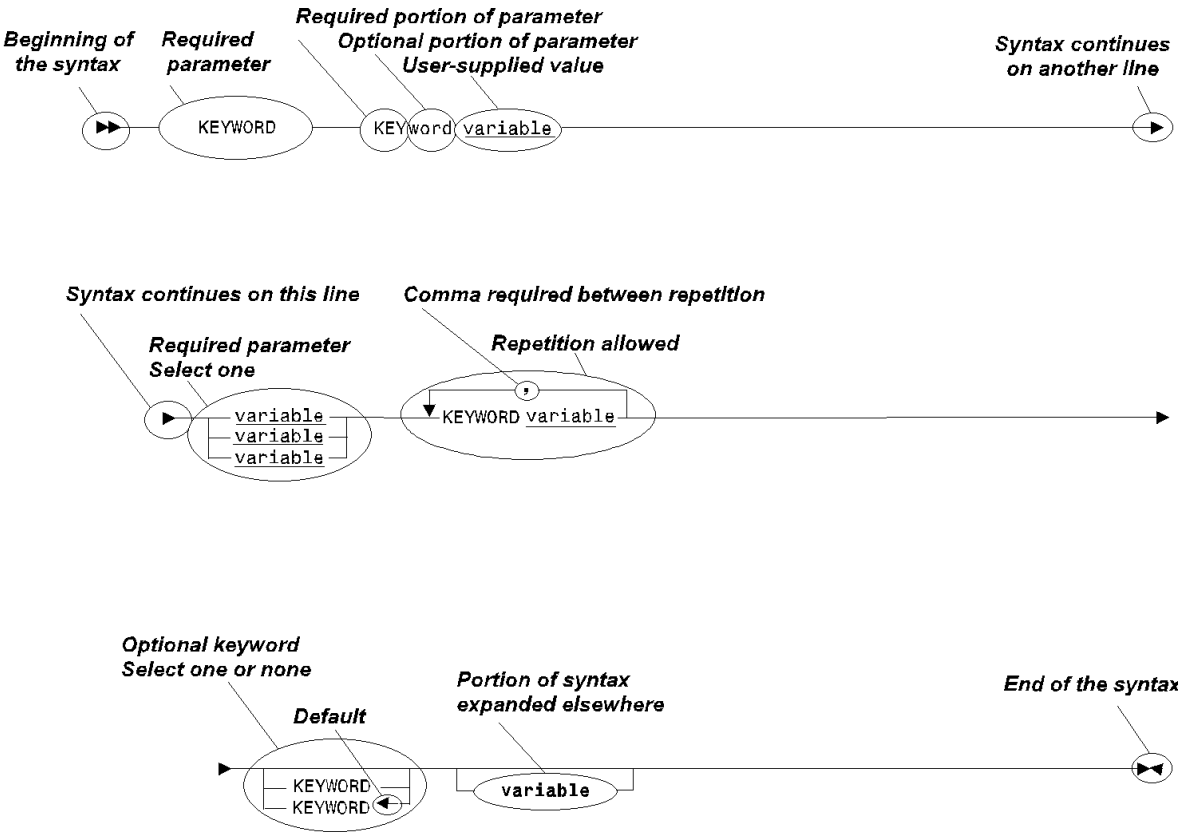
- *CA-IDMS Database Administration*
- *CA-IDMS Messages and Codes*
- *CA-IDMS System Generation*
- *CA-IDMS System Operations*
- *CA-IDMS System Tasks and Operator Commands*
- *CA-IDMS Utilities*

Understanding Syntax Diagrams

Look at the list of notation conventions below to see how syntax is presented in this manual. The example following the list shows how the conventions are used.

UPPERCASE OR SPECIAL CHARACTERS	Represents a required keyword, partial keyword, character, or symbol that must be entered completely as shown.
lowercase	Represents an optional keyword or partial keyword that, if used, must be entered completely as shown.
underlined lowercase	Represents a value that you supply.
←	Points to the default in a list of choices.
lowercase bold	Represents a portion of the syntax shown in greater detail at the end of the syntax or elsewhere in the document.
	Shows the beginning of a complete piece of syntax.
	Shows the end of a complete piece of syntax.
	Shows that the syntax continues on the next line.
	Shows that the syntax continues on this line.
	Shows that the parameter continues on the next line.
	Shows that a parameter continues on this line.
	Shows a required parameter.
	Shows a choice of required parameters. You must select one.
	Shows an optional parameter.
	Shows a choice of optional parameters. Select one or none.
	Shows that you can repeat the parameter or specify more than one parameter.
	Shows that you must enter a comma between repetitions of the parameter.

Sample Syntax Diagram



Chapter 1. Introduction

1.1 New Install and Installation Guide Features 1-3
1.2 Combined Installation Manual 1-4
1.3 CA-IDMS Installation Types 1-7
1.3.1 Complete Base Installation 1-7
1.3.2 Upgrade Installation 1-8
1.3.3 Add-On Installation 1-8
1.4 Using the New CICS Interface 1-10
1.4.1 Streamlined CICS Interface Installation 1-10

1.1 New Install and Installation Guide Features

Several new features are available in CA-IDMS Release 15.0, the installation procedures, and the installation guide, to simplify and aid you while installing and maintaining the CA-IDMS software.

The installation guide contains the installation and maintenance procedures for all of the CA-IDMS product family. Detailed information can be found in 1.2, “Combined Installation Manual” on page 1-4.

Also, there is a detailed discussion of the three *types* of installs available. See 1.3, “CA-IDMS Installation Types” on page 1-7, for more information.

Finally, with Release 15.0 of CA-IDMS, the IDMSINTC CICS interface can be installed and maintained using SMP/E. For more information, see 1.4, “Using the New CICS Interface” on page 1-10.

1.2 Combined Installation Manual

In Release 15.0 of CA-IDMS product family, the installation procedures for CA-IDMS, CA-IDMS Tools, and CA-Endevor/DB are merged into a single installation manual. These product lines and their respective products are listed below:

- CA-IDMS:
  - CA-ADS
  - CA-ADS ASF Option
  - CA-ADS/BATCH
  - CA-IDMS/APPC
  - CA-IDMS/CICS Support
  - CA-IDMS/Culprit™
  - CA-IDMS/DB
  - CA-IDMS/DBCS Option
  - CA-IDMS/DBOMP Transparency
  - CA-IDMS/DC
  - CA-IDMS/DDS (Distributed Database System)
  - CA-IDMS/Dictionary Loader
  - CA-IDMS/DLI Transparency
  - CA-EDP-Auditor™
  - CA-ICMS
  - CA-OLQ™
  - CA-IDMS/Performance Monitor
  - CA-IDMS/Presspack
  - CA-IDMS/SQL Option
  - CA-IDMS/Server
  - CA-IDMS/Total Transparency
  - CA-IDMS/UCF
  - CA-IDMS/VSAM Transparency
  - Unicenter TNG CA-IDMS/Agent
  - CA-VTX/PRESTEL
  - CA-VTX/TELETEL
  - CA-IDMS TP Option for CICS
  - CA-IDMS TP Option for INTERCOM

  - CA-IDMS TP Option for SHADOW
  - CA-IDMS TP Option for TASKMASTER
  - CA-IDMS TSO Interface
- CA-IDMS Tools:
  - CA-IDMS/ADS Alive
  - CA-IDMS/ADS Trace
  - CA-IDMS/Database Extractor
  - CA-IDMS/DB Analyzer
  - CA-IDMS/DB Audit
  - CA-IDMS/DB Reorg
  - CA-IDMS/DC SORT
  - CA-IDMS/Dictionary Migrator
  - CA-IDMS/Dictionary Migrator Assistant
  - CA-IDMS/Dictionary Module Editor
  - CA-IDMS/Dictionary Query Facility
  - CA-IDMS/DML Online
  - CA-IDMS/Enforcer
  - CA-IDMS/Journal Analyzer
  - CA-IDMS/Log Analyzer
  - CA-IDMS/Master Key
  - CA-IDMS/Online Log Display
  - CA-IDMS/SASO
  - CA-IDMS/Schema Mapper
  - CA-IDMS/Task Analyzer
- CA-Endevor/DB

While the installation manual is consolidated, the installation procedures are separate. The CA-IDMS core products installation **must be** completed before installing CA-Endevor/DB or the CA-IDMS Tools products.

Example: Suppose you are installing CA-IDMS/DB, CA-IDMS/DC, CA-IDMS/Performance Monitor, CA-IDMS/DML Online, and CA-Endevor/DB.

1. Install the core CA-IDMS products (CA-IDMS/DB, CA-IDMS/DC, CA-IDMS/Performance Monitor), using:
   a. Chapter 2, “Installation Prerequisites”
   b. Chapter 3, “Getting Started”

c. Chapter 4, “Installation Process”

d. Chapter 5, “VARBLIST Variables”

e. Chapter 6, “CA-IDMS Installation Jobs”

f. Chapter 7, “CA-IDMS Post-Installation Tasks”

2. Install the CA-IDMS Tools products (CA-IDMS/DML Online), using:

a. Chapter 3, “Getting Started”

b. Chapter 4, “Installation Process”

c. Chapter 5, “VARBLIST Variables”

d. Chapter 8, “CA-IDMS Tools Installation Jobs.”

e. Chapter 9, “CA-IDMS Tools Post-Installation Tasks”

3. Install CA-Endevor/DB, using:

a. Chapter 3, “Getting Started”

b. Chapter 4, “Installation Process”

c. Chapter 5, “VARBLIST Variables”

d. Chapter 10, “CA-Endevor/DB Installation Jobs”

e. Chapter 11, “CA-Endevor/DB Post Installation Tasks.”

1.3 CA-IDMS Installation Types

From a base-level install tape, you can perform one of the following mutually exclusive installations:

- Complete Installation
- Upgrade Installation
- Add-On Installation

	Complete Installation	**Upgrade Installation**	**Add-On Installation**
Creates New SMP/E Environment	Yes	Yes	No
Allocates/Formats Database Areas	Yes	No	Depends on product
Updates Existing Database Areas	No	Yes	Depends on product
Requires Previous SMP/E Base Install	No	No	Yes
Prior SMP/E Install Must Be Same Genlevel	N/A	No	Yes
Base Tape Required	Yes	Yes	Yes
Available on Maintenance Tape	No	No	No

Base tapes are not mass mailed; they are available only on individual order. Since prior-genlevel base tapes are expired, only the latest genlevel base tape can be ordered.

1.3.1 Complete Base Installation

Any site may execute a Complete Base Installation, although it is best suited for sites that have never installed CA-IDMS Release 12.0 or later. A Complete base installation:

- Creates a new SMP/E environment
- Creates new IDMS software in the SMP/E environment
- Allocates, formats, and initializes new dictionaries
- Creates new DMCL and DB name table load modules
- Creates a new DC system

A complete installation can be performed for CA-IDMS, CA-IDMS Tools, or CA-Endevor/DB.

1.3.2 Upgrade Installation

If you have already installed CA-IDMS Release 12.0 or later, you may choose to perform an Upgrade Installation. An Upgrade installation updates the CA-IDMS environment with the latest software and operates with your existing dictionaries and database files. It upgrades the products currently installed.

An Upgrade installation does not allocate, format, or initialize database or dictionary files. It *does* update dictionaries with new entities, such as:

- Messages and Codes
- System Records
- IDMS Protocols
- System Classes and Attributes
- Built-in Functions
- CA-IDMS Reports
- CA-IDMS Load Modules

A new SMP/E environment is allocated and populated. New software is created in the loadlibs. The SVC, startup module, and CICS interface load modules *must* be created from the new software.

Load modules, such as the DB name table and the DMCL, that are created by punching and linking source from your dictionaries do not need to be recreated.

An Upgrade install ignores allocation and page range parameters for database files. When you specify the customized values in VARBLIST, you do not have to modify the defaults for these items.

Note: Upgrade installations are valid for Release 12.0 and later CA-IDMS systems.

1.3.3 Add-On Installation

An Add-On installation allows you to install additional CA-IDMS or CA-IDMS Tools products into an existing CA-IDMS environment. Add-On installations must be performed on same-genlevel systems; bring your system up to the current service pack level by installing all necessary maintenance before adding the new product.

In VARBLIST, specify **INSTALL** for the product(s) you are adding, and specify *NO* for all other products.

- For CA-IDMS, an Add-On installation is not explicitly coded in the VARBLIST. CAISAG assumes this is an Add-On install when the variable, **CA-IDMS/DB=NO**. See 5.2.19, “Add-On Installation Variables” on page 5-11 for variables specifically related to an Add-On installation.

- For CA-IDMS Tools, determine if any products in the CA-IDMS Tools family have been installed, and set the WIADDON variable accordingly. See 1.2, “Combined Installation Manual” on page 1-4 for a list of CA-IDMS Tools products.
- There is no Add-On install for CA-Endevor/DB since it is a single product.

Depending on the product, some datasets may be allocated. For example, if you are installing CA-IDMS/SQL Option, datasets are allocated. VARBLIST parameters pertaining to space allocations and page ranges for other components ignored.

1.4 Using the New CICS Interface

In Release 15.0, the IDMSINTC CICS interface module is now delivered as an object module and is installed using SMP/E and can be maintained using SMP/E. Most users will be able to take advantage of this improved delivery method, although an earlier earlier version of the interface is still available.

1.4.1 Streamlined CICS Interface Installation

This feature simplifies the SMP/E installation of CA-IDMS by modifying the manner in which the IDMS interface and the various Transparencies for CICS are provided and installed, while continuing to allow the interface to be tailored to a particular environment.

The new interface is tailored using parameters specified during the installation process. These parameters are stored in an options module which is assembled during the installation process. The generated installation job stream links this options module with the appropriate distributed object decks under SMP/E control. Subsequent maintenance of the CICS interface and transparencies is accomplished by applying standard APARs through SMP/E.

The new interface requires CICS 4.1.0 or higher. If you are running an earlier version of CICS, you must continue to use the existing CICS interface which is delivered under the name INTCR141. Refer to the *CA-IDMS System Operations* manual for more information about the INTCR141 interface. The new CICS interface is tailored through the CICSOPT macro which an installation job assembles using parameters supplied during the installation process. The parameters for this macro are identical to those for the old IDMSINTC macro. For details, refer to Appendix E, “CICS Macros.”

In prior releases, if you extensively altered the existing CICS interface through source changes, you still have this capability because the new interface is delivered in source and object form. However, you will be unable to use SMP/E for subsequent maintenance. If it is necessary to modify the new interface to meet the requirements of your site, the source is contained in two macros: IDMSINTC and IDMSESC. CICS Interface Enhancements Release 15.0 provides the following functional enhancements to the CICS interface:

- TPNAME default
- Detection of CWADISP conflict
- Identification of the originating interface in messages
- Limits on the number of CICS users accessing IDMS through a particular interface
- Dynamic routing assistance
- OPTI exit support for SQL

Chapter 2. Installation Prerequisites

2.1 Hardware Requirements 2-4
2.1.1 Mainframe 2-4
2.1.1.1 CPU Requirements 2-4
2.1.2 Disk Drives 2-4
2.1.2.1 Disk Space 2-4
2.1.3 Tape Drive 2-5
2.2 Software\Interface Requirements 2-6
2.2.1 Operating System 2-6
2.2.2 SVC 2-6
2.2.3 Unicenter TNG Framework for OS/390 2-6
2.2.3.1 CA LMP Keys 2-7
2.2.4 VTAM 2-7
2.2.5 CICS 2-7
2.2.6 Libraries 2-8
2.2.7 SMP/E 2-8

This chapter reviews hardware, software and interface requirements for installing CA-IDMS.

2.1 Hardware Requirements

2.1.1 Mainframe

The following mainframes are supported, as well as plug-compatible systems:

- 30xx
- ES/9000
- S/390
- zSeries 900 enterprise server

2.1.1.1 CPU Requirements

The default configuration of CA-IDMS Release 15.0 requires a 6 Mb region for execution. Region requirements depend upon your specific system configuration.

2.1.2 Disk Drives

The following DASD are supported:

- 3375
- 3380
- 3390
- FBA

2.1.2.1 Disk Space

Disk space requirements are estimated in cylinders unless specified otherwise. The estimates are good approximations for all device types (i.e., 3380 or 3390).

Dataset	
SMP/E Datasets	86
Target Libraries	64
Distribution Libraries	180
Indirect Libraries	123
Database Files	200
CA-IDMS Tools	3400 Tracks
CA-Endevor/DB	360 Tracks
TOTAL	902

2.1.3 Tape Drive

One tape drive is required; the installation tape is available in reel, 3480- or 3490-cartridge format.

The installation process requires blank tapes when you select:

- A backup of the SMP/E environment
- A backup of the SYSTEM and SYSDIRL data dictionary
- A backup of the installed CA-IDMS DC/UCF system
- A journal dump/restore is requested during the execution of the demonstration job stream

2.2 Software\Interface Requirements

2.2.1 Operating System

The following operating systems are supported:

- OS/390 Release 2.5 or higher
- z/OS Release V1R1 or higher

2.2.2 SVC

CA-IDMS uses a Type 1 SVC to communicate between address spaces. Within CAIRIM, SVC numbers 172-176 are reserved for CA-IDMS; other valid user SVC numbers are 200-255.

The SVC number you plan to use must be selected prior to installation. The SVC load module, IGC*xxx* (where *xxx* is the SVC number), is assembled and linked by SMP/E during the installation process.

2.2.3 Unicenter TNG Framework for OS/390

The following Unicenter TNG Framework for OS/390 services at level B5 or higher are required. They are on the Unicenter TNG Framework for OS/390 media and their installation is described in the *Unicenter TNG Framework for OS/390 Getting Started Guide*.

Note: Release requirements for Unicenter TNG Framework for OS/390 may become dated or obsolete. Computer Associates recommends staying current with your Unicenter TNG Framework for OS/390 maintenance.

Service	Description
CAIRIM (Resource Initialization Manager)	Eliminates the need for user SVCs and other installation requirements commonly encountered when installing systems software by providing a common driver for a collection of dynamic initialization routines. CAIRIM installs the CA-IDMS SVC.
CA LMP (License Management Program)	Automated software tracking and validation service. Required by all CA-IDMS products. Note that CA LMP is a subcomponent of, and is installed along with, CAIRIM.

Service	Description
CAISSF (Standard Security Facility)	Allows CA-solutions to offer standardized security interfaces without regard for the particular needs of underlying access control software. CAISSF invokes CA-IDMS system security features. Note that CAISSF is a subcomponent of, and is installed along with, CAIRIM.
CA-C Runtime (CA-C Runtime System)	Required by the CA-ADS compiler, CA-C is the C language runtime engine for C language-based CA-solutions.
CAICCI (Common Communication Interface)	Allows your software solutions to work together across platforms, making your software more powerful. CAICCI is required by CA-IDMS/DDS (Distributed Database System) and CA-IDMS/Server.
CAIENF (Event Notification Facility)	Insulates software solutions from changes in the operating system and environment. CAIENF is required by CA-IDMS/DDS (Distributed Database System) and CA-IDMS/Server.

Since various Unicenter TNG Framework for OS/390 services are also required by other Computer Associates products, the components you need for CA-IDMS may be installed at your site. Check with your systems group.

2.2.3.1 CA LMP Keys

The CA-IDMS product family requires CA LMP keys. CA LMP keys are valid across product releases but not across CPUs. If you are installing a new product, or you are installing on a different CPU, CA LMP keys must be installed. For CA LMP key installation information, refer to 3.3, “CA LMP Keys” on page 3-6.

2.2.4 VTAM

A VTAM application ID (APPLID) is required for the DC/UCF system. The following could be used to assign an APPLID of IDMSDC:

```
IDMSDC APPL AUTH=(ACQ,NOPASS,NVPACE,NOTSO,NOPO)
```

2.2.5 CICS

The CA-IDMS installation does not require CICS.

If your site wants to access CA-IDMS from CICS, CA-IDMS/CICS Support interface supports CICS Release 4.1 and higher including CICS Transaction Server 1 and above. To implement the interface, CA-IDMS's Central Version (CV) requires the exclusive

use of a fullword in the CICS CWA. Typically, only the CICS systems programming staff assigns fullwords in the CWA.

2.2.6 Libraries

The installation and operation of some products in the CA-IDMS product line require the following libraries:

- The OS/390 system macro library (SYS1.MACLIB) is required by CA-IDMS/DB.
- A COBOL compiler runtime library and COBOL subroutine library are required to install the Commonweather demonstration database.
- CICS libraries are required by CA-IDMS/CICS Support and CA-IDMS TP Option for CICS.
- If your site uses TCAM or BTAM, CA-IDMS/DC requires standard TP libraries.
- The CA-C runtime library is required by the CA-ADS compiler. The CA-C runtime modules are typically found in the Unicenter TNG Framework for OS/390 library, CAILIB.
- SYS1.AMODGEN or SYS1.MODGEN is required, if you are installing CA-IDMS TP Option for CICS or CA-IDMS/VSAM Transparency and SYS1.MACLIB does not contain IKJTCB and IEFJSSVT.

2.2.7 SMP/E

SMP/E Release 1.8 or higher is required. CA-IDMS cannot be installed without SMP/E.

Chapter 3. Getting Started

3.1 Overview of Install Steps 3-4
3.2 Review Cover Letters and PMLs 3-5
3.3 CA LMP Keys 3-6
3.4 Remove USERMODs 3-9

This chapter discusses the information required to install CA-IDMS, CA-IDMS Tools and CA-Endevor/DB

3.1 Overview of Install Steps

The following list summarizes the steps involved in the CA-IDMS, CA-IDMS Tools and CA-Endevor/DB installation process. Review this list before you begin the installation process.

1. Review the installation package cover letter and any PMLs included with the installation package for pertinent installation information.
2. Add the CA LMP execution keys to CAIRIM.
3. Review the CA-IDMS, CA-IDMS Tools, and CA-Endevor/DB system requirements. Be sure all requirements are met before starting the installation process.
4. Remove USERMODs containing JCLIN.
5. Load the SAMPJCL library from the installation tape to a disk.
6. Customize members in the SAMPJCL library.
7. Execute the DOWNLOAD and CAISAG jobs to generate the customized installation JCL.
8. Review and execute the installation JCL generated by CAISAG.
9. Complete the appropriate post-installation tasks.

The remainder of this chapter describes the first three steps in detail.

3.2 Review Cover Letters and PMLs

Review cover letters and Product Maintenance Letters (PMLs) in your CA-IDMS installation package before initiating the installation process. These letters contain additional information not found in this guide.

Warning: If the cover letter or PMLs' instructions conflict with this manual, the directions in the letters supersede this guide.

3.3 CA LMP Keys

Each family of products (CA-IDMS, CA-IDMS Tools, and CA-Endevor/DB) requires CA LMP keys.

CA LMP keys are valid across product releases but not across CPUs. If you are running a prior release of the product(s) and you are installing on the same CPU, you can skip this section. If you are installing a new product, or you are installing on a different CPU, you must install CA LMP keys.

Examine the CA LMP Key Certificates you received with your installation package. You should have a key certificate for every product you are installing that requires one. Appendix A, "CA-IDMS Product List" contains a list of products.

Key Certificates contain the following information:

Fields	Descriptions
Product Name	The trademarked or registered name of the Computer Associates product licensed for the designated site and CPUs.
Supplement	The reference number of your license for the particular product, in the format nnnnnn - nnn. This format differs slightly inside and outside North America, and in some cases may not be provided at all.
Expiration Date	The date (MONTH dd, yyyy as in JANUARY 15, 2002) your license for this product expires.
Technical Contact	The name of the technical contact at your site, who is responsible for the installation and maintenance of the designated product. This is the person to whom CA addresses all CA LMP correspondence.
MIS Director	The name of the director of MIS, or the person who performs that function at the site. If the title but not the individual's name is indicated on the Certificate, please supply the actual name when correcting and verifying the Certificate.
CPU Location	The address of the building where the CPU is installed.

Fields	Descriptions
Execution Key	An encrypted code required by CA LMP for product initialization. During installation, it is referred to as the LMP code.
Product Code	A two-character code that corresponds to this particular product.
CPU ID	The code that identifies the specific CPU for which installation of your product is valid.

If your site is already running software from Computer Associates, your systems group probably is already running a CAS9 proc (cataloged procedure). If so, add the CA LMP keys for the products you are installing to the CA LMP keys already being used.

The CAS9 proc runs the CAIRIM program which uses a KEYS DD statement. The KEYS DD statement should point to your CA LMP keys. The KEYS DD statement usually points to a member named KEYS in a PARMLIB or OPTLIB. Each line contains the information from one key certificate. If the information does not fit on one line, code a hyphen in column 72 for a continuation character.

The parameter structure for member KEYS is presented below:

Note: The parentheses in the parameter list are necessary.

```
PROD(pp) DATE(ddmmmyy) CPU(tttt-mmmm/ssssss) LMPCODE(kkkkkkkkkkkkkkkk)
```

Parameter	Definition
pp	Required. The two-character product code. For any given CA LMP software solution, this code agrees with the product code already in use by the CAIRIM initialization parameters for earlier genlevels of the product.
ddmmmyyy	The CA LMP licensing agreement expiration date.
tttt-mmmm	Required. The CPU type and model (for example: 3090-600) on which the CA LMP software solution is to run. If the CPU type and/or the model require less than four characters, blank spaces are inserted for the unused characters.
ssssss	Required. The serial number of the CPU on which the CA LMP software solution is to run.

Parameter	Definition
kkkkkkkkkkkkkkkk	Required. The execution key needed to run the CA LMP software solution. This CA LMP execution key is provided on the Key Certificate shipped with each CA LMP software solution.

Example: This is an example of a control card for the CA LMP execution software parameter.

```
PROD(BD) DATE(15JAN02) CPU(3090-600 /370623) LMPCODE(52H2K06130Z7RZ06)
```

Note: This is an example only, the CA LMP execution key is invalid. For more information about CA LMP execution key definition, see the *Unicenter TNG Framework for OS/390 Getting Started Guide*.

3.4 Remove USERMODs

The CA-IDMS Tools and CA-Endevor/DB installations link several CA-IDMS modules; USERMODs referencing these CA-IDMS modules must be removed prior to installing these products or they cannot be removed later. Therefore, if you plan to install CA-IDMS Tools or CA-Endevor/DB, either:

- Create the USERMOD for testing the CA-IDMS software environment; RESTORE the USERMOD before installing CA-IDMS Tools or CA-Endevor/DB
- Delay creating USERMODs containing JCLIN statements until after these products are installed

These CA-IDMS modules are included in the CA-IDMS Tools and CA-Endevor/DB installs:

IDMS Module Name	CA-IDMS Tools	CA-Endevor/DB
IDDPRST	X	
IDDSFEBT	X	
IDDSFEDC	X	
IDDSPIDD	X	
IDMS	X	X
IDMSBALI	X	X
IDMSCALC	X	
IDMSCANC	X	
IDMSCHPT	X	
IDMSINTC	X	
IDMSCOMP	X	
IDMSDATE	X	
IDMSDCOM	X	
IDMSFSED	X	
IDMSUBPT	X	
RHDCDATE	X	
RHDCGNPM	X	
RHDCUXIT	X	

Example: You have a USERMOD that creates a load module called ABC, and the link (JCLIN) for ABC contains the module IDMS. CA-IDMS Tools and CA-Endevor/DB contain links including the IDMS module. If you install either product line before removing ABC USERMOD, you cannot remove the USERMOD.

Chapter 4. Installation Process

4.1 Overview 4-3
4.2 Download SAMPJCL 4-4
4.3 Download CAISAG Program and Skeleton 4-5
4.4 Customize the CAISAG Input 4-6
4.4.1 JOBCARD 4-6
4.4.2 PASSWORD 4-6
4.4.3 SETUP 4-6
4.4.4 SETUPBK 4-7
4.4.5 VARBLIST 4-7
4.5 Run CAISAG 4-9

4.1 Overview

The installation procedures described in this chapter, apply to all three install types: Add-On, Complete Base and Upgrade. The installation procedure is based on the Specify and Generate (CAISAG) program. When this program is executed, it reads installation variables you specified, reads a prototype JCL skeleton containing default values, and generates customized installation jobs for your site. The CA-IDMS installation process is designed for database administrators.

The installation jobs utilize SMP/E, an IBM utility, to install the software. The installation jobs allocate a new SMP/E environment for CA-IDMS automatically and CA-IDMS is segregated from other products. This way only the CA-IDMS SMP/E environment is affected. You do not affect the OS/390 system by running SMP/E any more than you would affect SYS1.LINKLIB by running the linkage editor. An exhaustive knowledge of SMP/E is not required for this install, because the JCL to run SMP/E is generated when you run the CAISAG program; a general understanding of SMP/E is recommended to maintain the system. See Appendix D, “SMP/E” for more information about SMP/E.

CA-IDMS Tools and CA-Endevor/DB must be installed into the same SMP/E environment (CSI and target zone) as CA-IDMS.

4.2 Download SAMPJCL

The first step in the installation process is downloading the SAMPJCL library from the install tape using the JCL specified in the cover letter.

The downloaded SAMPJCL library *always* contains the following members:

Member	Description
CAISAG	JCL to run the Specify and Generate program
DOWNLOAD	JCL to download additional CAISAG input
JOBCARD	Jobcard used as CAISAG input
PASSWORD	Member containing passwords, used as CAISAG input
SETUP	Member containing /*SETUP cards, used as CAISAG input
VARBLIST	CAISAG variables; see Appendix C, “VARBLIST”

CAUTION:
Since these member names are used in each install procedure for CA-IDMS, CA-IDMS Tools, and CA-Endevor/DB, separate SAMPJCL libraries must be allocated for each product line.

4.3 Download CAISAG Program and Skeleton

The DOWNLOAD members for each install are similar.

```
//*********************************************************************
//*                                                                   *
//*   THIS JOB DOWNLOADS THE 'SPECIFY AND GENERATE' PROGRAM           *
//*   AND ITS SKELETON FROM THE INSTALLATION TAPE.                    *
//*                                                                   *
//*********************************************************************
//PROGRAM  EXEC PGM=IEWL,PARM='LET,LIST,SIZE=(524288,65536),NCAL'
//*    EXPECTED RETURN CODE:  00
//SYSPRINT DD SYSOUT=*                                <=== SYSOUT CLASS
//SYSUT1   DD SPACE=(CYL,(5,1)),UNIT=sysda            <=== WORK UNIT
//SYSLIN   DD DISP=(OLD,PASS),UNIT=tape,              <=== TAPE UNIT
//            DSN=CAI.volser.FILEnn,LABEL=nn,
//            VOL=(PRIVATE,RETAIN,SER=volser)
//SYSLMOD  DD DSN=IDMS.R150.CAISAG.LOADLIB,           <=== DSNAME
//            SPACE=(CYL,(2,,1)),UNIT=sysda,          <=== DISK UNIT
//            VOL=SER=idms01,                         <=== DISK VOLSER
//            DISP=(NEW,CATLG,DELETE),
//            DCB=(RECFM=U,LRECL=15476,BLKSIZE=15476)
/*
//SKELETON EXEC PGM=IEBGENER
//*    EXPECTED RETURN CODE:  00
//SYSPRINT DD SYSOUT=*                                <=== SYSOUT CLASS
//SYSIN    DD DUMMY
//SYSUT1   DD DSN=CAI.volser.FILEnn,
//            DISP=(OLD,PASS),UNIT=tape,              <=== TAPE UNIT
//            VOL=(PRIVATE,SER=volser),LABEL=nn,
//            DCB=(RECFM=F,LRECL=8000,BLKSIZE=8000)
//SYSUT2   DD DSN=IDMS.R150.CAISAG.SKELETON,          <=== DSNAME
//            SPACE=(TRK,(21,1)),UNIT=sysda,          <=== DISK UNIT
//            VOL=SER=idms01,                         <=== DISK VOLSER
//            DISP=(NEW,CATLG,DELETE),
//            DCB=(RECFM=F,LRECL=8000,BLKSIZE=8000)
/*
//
```

Customize the DOWNLOAD member and copy the SETUP and JOBCARD members into it. If necessary, change the job class to one suitable for a tape job. Run the job to download the CAISAG utility and the skeleton JCL.

4.4 Customize the CAISAG Input

4.4.1 JOBCARD

The default JOBCARD member is shown below:

```
//CAIIDMS JOB (ACCOUNT INFORMATION),'                    ',
//    CLASS=A,NOTIFY=USER0001,MSGCLASS=X
```

Customize the JOBCARD member for your site, suitable for long-running jobs without tape mounts. This member is copied into the top of each generated job. When tape mounts are required, CAISAG modifies the job class.

4.4.2 PASSWORD

The default PASSWORD member has the following format:

```
PASSWORD 12345678
PASSWORD 12345678
PASSWORD 12345678
PASSWORD 12345678
⋮
PASSWORD 12345678
PASSWORD 12345678
PASSWORD 12345678
```

Use the Product Authorization Sheets shipped with the tape to customize the PASSWORD member. Put one password on each line, replacing the '12345678'. Extra lines can be deleted.

4.4.3 SETUP

This is the default SETUP member:

```
//*
```

The SETUP member is copied into the beginning of each job requiring a tape mount of the installation tape. If your site uses /*SETUP cards, customize the SETUP member with /*SETUP cards appropriate for the install tape.

CAUTION:
Do not delete the SETUP member.

4.4.4 SETUPBK

This is the default SETUPBK member:

```
//*
```

The SETUPBK member is copied into the beginning of each job requiring a backup tape to be mounted. If your site uses /*SETUP cards, customize the SETUPBK member with the /*SETUP cards appropriate for the backup tapes.

CAUTION:
Do not delete the SETUPBK member.

4.4.5 VARBLIST

The VARBLIST contains a list of variables used by the Specify and Generate program to generate customized JCL for your site. For each variable, supply the appropriate value for your site. The format for each variable:

```
variable-name = value       optional comment
```

When coding the VARBLIST member, keep in mind:

- An asterisk in column 1 denotes a comment.
- Blank lines may be used for readability.
- Values containing blanks must be enclosed in single quotes.
- Null values are indicated by two single quotes.
- All values are translated to upper case, unless they are enclosed in single quotes.

Many values in VARBLIST begin and end with a **@**. If the string inside the **@** characters is a previously-defined variable, CAISAG replaces the string and the **@** characters with the variable's value. For example:

```
PREFIX  = IDMS.R150
LOADLIB = @PREFIX@.LOADLIB
```

generates the same JCL as:

```
PREFIX  = IDMS.R150
LOADLIB = IDMS.R150.LOADLIB
```

If a variable is specified more than once, CAISAG uses the last value.

See Chapter 5, “VARBLIST Variables” for detailed information regarding the variables for CA-IDMS, CA-IDMS Tools and CA-Endevor/DB VARBLISTs. Copies of the VARBLISTs are available in Appendix C, “VARBLIST.”

CAUTION:
The variables in the VARBLIST member are subject to change change from one genlevel to the next. Using a VARBLIST from a prior genlevel can cause unpredictable results.

4.5 Run CAISAG

A sample CAISAG member is shown below:

```
//*********************************************************************
//*                                                                   *
//*   NOTE:  BEFORE RUNNING THIS JOB, CUSTOMIZE THE FOLLOWING         *
//*          MEMBERS:                                                 *
//*              JOBCARD   (CREATE A LEGAL JOBCARD FOR YOUR SITE)     *
//*              PASSWORD  (USE THE PRODUCT AUTHORIZATION SHEETS)     *
//*              SETUP     (UPDATE IF YOUR SITE USES /*SETUP CARDS)   *
//*              SETUPBK   (SETUP CARDS FOR BACKUPS)                  *
//*              VARBLIST  (ALL OTHER INSTALLATION VARIABLES)         *
//*                                                                   *
//*   THIS JOB RUNS THE 'SPECIFY AND GENERATE' PROGRAM.  THE          *
//*   PROGRAM READS YOUR SITE-SPECIFIC VALUES FROM THE MEMBERS        *
//*   SPECIFIED ABOVE, AND GENERATES CUSTOMIZED INSTALLATION JCL.     *
//*   THE VARBLIST, PASSWORD, JOBCARD AND SETUP MEMBERS MUST ALL      *
//*   BE AVAILABLE IN ONE LIBRARY.                                    *
//*                                                                   *
//*   THIS JOB CAN BE RUN REPEATEDLY TO SEE THE EFFECTS THAT          *
//*   DIFFERENT VALUES IN THE VARIABLE LIST HAVE ON THE GENERATED     *
//*   JCL.                                                            *
//*                                                                   *
//*   CUSTOMIZE THE JCL BELOW TO POINT TO THIS LIBRARY (SAMPJCL)      *
//*   AND TO THE DATASETS ALLOCATED IN THE DOWNLOAD JOB.              *
//*                                                                   *
//*   THIS JOB CREATES NEW MEMBERS IN THE SAMPJCL LIBRARY.           *
//*   SEE COMMENTS IN THE VARBLIST MEMBER FOR MORE INFORMATION.       *
//*                                                                   *
//*********************************************************************
//CAISAG   EXEC PGM=CAISAG
//STEPLIB  DD DISP=SHR,
//            DSN=IDMS.R150.CAISAG.LOADLIB        <=== DOWNLOAD DSNAME
//GENJCL   DD DISP=(NEW,PASS),DSN=&&GENJCL,
//            SPACE=(TRK,(15,10)),UNIT=SYSDA,      <=== WORK UNIT
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=8880)
//RECORD   DD DCB=BLKSIZE=80,SYSOUT=*              <=== SYSOUT
//MESSAGE  DD DCB=BLKSIZE=80,SYSOUT=*              <=== SYSOUT
//SKELETON DD DISP=SHR,
//            DSN=IDMS.R150.CAISAG.SKELETON        <=== DOWNLOAD DSNAME
//SAMPJCL  DD DISP=SHR,
//            DSN=IDMS.R150.SAMPJCL                <=== YOUR SAMPJCL
/*
//IEBUPDTE EXEC PGM=IEBUPDTE,PARM=NEW,COND=(4,LT)
//SYSUT2   DD DISP=OLD,
//            DSN=IDMS.R150.SAMPJCL                <=== YOUR SAMPJCL
//SYSIN    DD DISP=(OLD,DELETE),DSN=&&GENJCL
//SYSPRINT DD SYSOUT=*                             <=== SYSOUT
/*
```

The CAISAG job generates customized installation jobs as new members in the SAMPJCL library.

CAISAG is a quick-running job. Choosing the appropriate installation values is an iterative process. After you customize VARBLIST and the other members, run CAISAG and browse the generated jobs. You can quickly correct a variable or two and refresh the generated JCL. VARBLISTs from previous installations are an invaluable tool for documenting your system.

CAUTION:
Do not submit any of the generated install jobs until you have verified the JCL in all the installation jobs.

Chapter 5. VARBLIST Variables

5.1 Common Variables 5-4
5.1.1 Member Name 5-4
5.1.2 Products 5-4
5.1.3 Global DMCL 5-4
5.2 CA-IDMS Variables 5-5
5.2.1 Upgrade 5-5
5.2.2 Tape Class 5-5
5.2.3 Disk VOLSER and Disk Contention 5-5
5.2.4 SMP/E Environment 5-6
5.2.5 CV Number 5-7
5.2.6 DC System Number 5-7
5.2.7 Free Storage 5-7
5.2.8 Access Method 5-8
5.2.9 Uppercase Terminal Support 5-8
5.2.10 Authorized Userid 5-8
5.2.11 Link WTOEXIT with Startup Module 5-8
5.2.12 Startup Module 5-8
5.2.13 Dynamic or Static PDE Support 5-9
5.2.14 Storage Protection 5-9
5.2.15 Backup SMP/E Environment 5-10
5.2.16 VTAM ID 5-10
5.2.17 CWA Displacement 5-11
5.2.18 CA-Culprit 5-11
5.2.19 Add-On Installation Variables 5-11
5.2.20 Page Size 5-11
5.2.21 Unicenter TNG Framework for OS/390 5-12
5.3 CA-IDMS Tools Variables 5-13
5.3.1 Upgrade 5-13
5.3.2 Create a CA-IDMS Tools Dictionary 5-13
5.4 CA-Endevor/DB Variables 5-14
5.4.1 Existing CA-Endevor/DB into a New SMP/E Environment 5-14
5.4.2 Upgrade CA-Endevor/DB 5-14
5.4.3 CA-Endevor/DB Message Library 5-14

This chapter assists you with installation decisions and provides detailed information for variables that must be specified in the VARBLIST file before running CAISAG. Each installation has its own unique VARBLIST; where appropriate, identical variable names are used in each VARBLIST. VARBLIST variables are heavily documented.

If this is a Complete installation of CA-IDMS, the dictionary and database file sizes are sufficient for generating a small system to test the installation. Files sizes used in an earlier installation are a better indication of the sizes needed for your environment.

5.1 Common Variables

These are common variables found in the VARBLIST for each product family.

5.1.1 Member Name

```
BEGINMEM=member-name
```

This variable establishes the member names for all the generated installation jobs. The installation jobs are created in the SAMPJCL library. The member-name is one to six alphanumeric characters. Two numbers are appended by CAISAG to create unique member names.

This feature allows you to create multiple versions of the installation jobs while maintaining a record of the original install.

Example: After installing CA-IDMS Release 15.0, you decide to install an additional CA-IDMS product. By changing the BEGINMEM value, a history is maintained of the original complete install as well as the new Add-On install.

5.1.2 Products

```
product-name=INSTALL
             NO ←
```

See Appendix A, “CA-IDMS Product List” for the products on the CA-IDMS install tape. To install a particular product, change its value in VARBLIST from NO to INSTALL and update the PASSWORD member with its password, if one was provided for the product on a Product Authorization sheet.

To access a DC/UCF system from CICS, you must install CA-IDMS/CICS Support, TP/CICS and CA-IDMS/UCF.

5.1.3 Global DMCL

```
GJGLDMCL=dmcl-name
```

This variable specifies the name of the global DMCL. If you are performing a CA-IDMS Complete installation, this specifies the new global DMCL; otherwise, it is the current global DMCL.

5.2 CA-IDMS Variables

5.2.1 Upgrade

```
GJUPGRAD=YES
         NO ←
```

An Upgrade install is available for CA-IDMS Release 12.0 and later, and:

- Provides enhanced CA-IDMS software for your existing system
- Creates a new SMP/E environment by allocating and populating new libraries
- May update database areas, but does not allocate or format database areas

The default is NO, indicating a Complete base install. For more information about Complete base installs, see 1.3.1, “Complete Base Installation” on page 1-7.

If you choose an Upgrade install, (YES):

- CAISAG ignores all parameters for database file allocation
- Specify the name of the existing global DMCL in the GJGLDMCL (5.1.3, “Global DMCL” on page 5-4) variable
- Additional products **cannot** be installed

CAUTION:
If you are running CA-IDMS Release 10.2 (or earlier), you cannot utilize the Upgrade install.

5.2.2 Tape Class

```
TAPCLASS=T ←
         x
```

The TAPCLASS variable specifies the job class to be used for jobs requiring a tape mount. For jobs requiring a tape mount, CAISAG automatically substitutes the tape class in the CLASS= parameter on the jobcard.

5.2.3 Disk VOLSER and Disk Contention

```
GJxxxSER=volume
```

This parameter allows you to specify the pack on which to allocate the dataset. For every dataset created by the install process, there is a corresponding variable designating the disk VOLSER. If you want the operating system to choose the pack, specify a null value (two single quotes) for the GJ*xxx*SER variable. Or you can use:

```
PACK     = ''
GJxxxSER = @PACK@
```

The GJxxxSER variable is useful for reducing or eliminating disk contention issues. Areas such as the DDLDCLOD, DDLDCLOG and DDLDCMSG areas are heavily used at runtime and should be assigned to separate DASD to reduce disk contention.

Journal file placement is important; for optimum performance allocate four journal files on separate DASD. Optimum allocation when you have two disk drives:

File Name	Disk 1	Disk 2
Journal 1	X	
Journal 2		X
Journal 3	X	
Journal 4		X

If there are three packs available, this is the optimum allocation:

File Name	Disk 1	Disk 2	Disk 3
Journal 1	X		
Journal 2		X	
Journal 3	X		
Journal 4			X

5.2.4 SMP/E Environment

```
GJNEWCSI=YES ←
         NO
```

This option specifies if CA-IDMS is installed in its own SMP/E environment (YES) or a shared SMP/E environment with other Computer Associates products (NO). An SMP/E environment consists of the CSI and its related datasets.

Computer Associates strongly recommends installing CA-IDMS into its own SMP/E environment. This isolates the CA-IDMS SMP/E environment from other SMP/E environments therefore easing maintenance. You can backup the CA-IDMS SMP/E environment regularly and restore the environment without disrupting other products. If the CSI becomes corrupted or out of sync with the CA-IDMS libraries, you must either:

- Restore the SMP/E environment from a backup
- Reinstall the software

Warning: Do not install CA-IDMS into a CSI containing CA-DISPATCH or a prior release of CA-IDMS.

5.2.5 CV Number

GJCVNUM=nnn

The CV number identifies the DC/UCF system to the SVC. Each SVC contains 256 slots used for communication between address spaces. The CV number specifies the slot used by a particular CV. Valid values for the CV number are 0 through 255. If two central versions are assigned the same CV number, they cannot execute concurrently.

5.2.6 DC System Number

GJDCSYS=nnnn

The DC system number uniquely identifies a DC/UCF system. DC system numbers:

- Valid values are 1 - 9999[1]
- Must be unique

The DC system number identifies the system being generated, modified, copied, or deleted. It is used in assembling RHDCPARM, and can be used in the S= parameter on the EXEC card in the startup deck (JCL to bring the system up) to override the DC system number assembled in RHDCPARM.

Note: [1] Many sites synchronize the DC system with the CV number (see 5.2.5, "CV Number") assigning a value from 1 to 255.

Warning: Do not assign DC system numbers 90 and 99. These system numbers are reserved for use by CA-IDMS during the installation process.

For more information about:

- RHDCPARM, see *CA-IDMS System Operations*
- DC System Number, see *CA-IDMS System Generation*

5.2.7 Free Storage

GJFRESTG=nnnn (default=1200)

The amount of storage, in K bytes returned (freed) to the operating system at DC/UCF startup time. The storage is freed for operating system use during DC/UCF startup operations. GJFRESTG must specify a positive integer storage amount.

GJFRESTG is passed to #DCPARM parameter FREESTG; for more information about #DCPARM or FREESTG, see the *CA-IDMS System Operations* guide.

Operating system abends due to insufficient memory during startup are likely the result of an inadequate amount of storage specified in the GJFRESTG variable.

The GJFRESTG variable also affects the amount of storage available to the operating system for extending the LSQA at runtime.

Note: If you are installing the Unicenter TNG CA-IDMS/Agent, add an additional 400K to the GJFRESTG value.

5.2.8 Access Method

```
GJACMETH=BDAM ←
         VSAM
```

BDAM is recommended since there is no advantage to using VSAM access method in OS/390.

If you use native VSAM, Release 15.0 could require additional storage for startup, compared to prior releases.

5.2.9 Uppercase Terminal Support

```
GJCASE=MIXED ←
       UPPER
```

CA-IDMS products support both uppercase only and mixed case installations. If site requirements prohibit displaying mixed-case characters, specify UPPER. This often occurs due to language issues, not hardware limitations.

5.2.10 Authorized Userid

```
GJAUTH=userid
```

The GJAUTH and GJAUTHPW variables should be left with null values unless you are doing an Upgrade or Add-On install and your dictionaries are secured.

5.2.11 Link WTOEXIT with Startup Module

```
GJWTOEX = YES ←
          NO
```

This variable specifies whether you want to include a WTOEXIT module when the system startup modules are created.

5.2.12 Startup Module

```
GJSTUP=modulename
```

The GJSTUP variable allows you to specify the name for an *additional* startup module. If no value is entered, only the default modules IDMSDC and IDMSDCU are created.

5.2.13 Dynamic or Static PDE Support

```
GJPDETYP=DYNAMIC ←
         STATIC
```

The GJPDETYP parameter specifies how CA-IDMS allocates program definition elements (PDEs) for Computer Associates-supplied maps, dialogs, and tables. DYNAMIC, the default, indicates the PDEs are allocated at runtime; their PROGRAM statements are not included in the system definition. STATIC designates the PDEs are allocated during system startup for Computer Associates-supplied maps, dialogs, and tables.

During the installation process, PROGRAM statements defining the maps, dialogs, and tables for a CA-IDMS product are *not* included in the System 99 definition. As a result, PDEs are allocated at startup for those programs requiring them and dynamically for those programs eligible for dynamic PDE allocation. For a CA-IDMS product using maps, dialogs, or tables, the PROGRAM statements are included in two separate dictionary modules:

- One module contains the required PROGRAM statements
- Another module contains the PROGRAM statements for the maps, dialogs, and tables; these module names are suffixed with -DYN.

To add these PROGRAM definitions to your system definition, add the appropriate INCLUDE statements for the dictionary modules containing the PROGRAM statements for the product-specific maps, dialogs, and tables and regenerate the system definition.

Example: The PROGRAM statements for the programs defining OLQ menu mode are stored in the dictionary module ONLINE-QUERY. The PROGRAM statements for the dialogs, maps, and tables defining OLQ menu mode are stored in the dictionary module ONLINE-QUERY-DYN.

For more information, see the *CA-IDMS System Generation* manual.

5.2.14 Storage Protection

```
STORPROT = YES ←
           NO
```

STORPROT variable specifies whether programs are added to the sysgen as PROTECT or NOPROTECT. This parameter appears in the CA-IDMS, CA-IDMS Tools and CA-Endevor/DB VARBLISTs.

If STORPROT=YES, all PROGRAM statements contain the PROTECT parameter and the CV SYSTEM statement includes a NOPROTECT parameter. This is the sysgen compiler default, and is a good choice for test systems.

When STORPROT=NO, all PROGRAM statements include the NOPROTECT parameter and the CV SYSTEM statement contains the PROTECT parameter. This is a good choice for a production system.

Regardless of the value chosen for STORPROT, all source members are in the DISTSRC library. At a later date, you can rerun the IDMSDDDL installation steps and replace the modules with the other versions.

Note: Appendix B of the *CA-IDMS System Generation* manual contains a list of source members and module names for CA-IDMS, CA-IDMS Tools and CA-Endevor/DB

For more information about the PROGRAM statement, refer to the *CA-IDMS System Generation* manual

5.2.15 Backup SMP/E Environment

```
GJSMPBKP=YES ←
         NO
GJSBTAPE=IDMS15 ←
         volser
GJINITT=YES ←
        NO
```

Computer Associates strongly recommends backing up the CA-IDMS SMP/E environment after you have completed the installation process and verified the environment. These variables simplify the process by generating the JCL for you.

- GJSMPBKP determines if the JCL for backing up the SMP/E environment is generated. YES, the default, generates the backup JCL.
- GJSBTAPE specifies the volume serial name for the tape. If the VOLSER is not specified, the operator may use a scratch tape.
- GJINITT indicates if the tape is to be initialized. YES, the default, creates an IEHINITT step to initialize the tape.

5.2.16 VTAM ID

```
GJVTAMID='' ←
         applid
```

GJVTAMID is only used for those sites installing CA-IDMS/DC. If left null, the default, the VTAM line is not installed. Otherwise, specify the name of the APPLID.

5.2.17 CWA Displacement

```
GJCWA=nn
```

As stated earlier, CA-IDMS CV requires exclusive use of a fullword in the CICS CWA. Specify the decimal displacement relative to zero from the start of the CWA.

5.2.18 CA-Culprit

The keywords and operands for CA-Culprit profile CSECT options are shown in Appendix F, "CA-Culprit Profile Options."

5.2.19 Add-On Installation Variables

```
GJNUMJRN=''
GJNWDMCL=''
```

When you install CA-IDMS for the first time, CA-IDMS/DB is a required product. Subsequent installs into the same SMP/E environment for additional products are known as Add-On installations.

For an Add-On install, specify NO for all the products except the one(s) you are installing. These variables pertain to Add-On installs only:

- GJNUMJRN specifies the number of disk journals defined to the system.
- GJNWDMCL indicates the new name for the DMCL.

For an Add-On install, the only datasets allocated are the indirect libraries (temporary download libraries) unless you are installing the ASF-OPTION or CA-IDMS/SQL Option which have their own databases.

For an Add-On install, Computer Associates recommends changing the value of the BEGINMEM variable to avoid overwriting your original installation jobs.

An Add-On install must be done from a base tape. Do not skip genlevels. Before executing an Add-On installation, apply all necessary maintenance to the original products to bring them up to the genlevel of the base tape you are using for the Add-On install.

5.2.20 Page Size

```
GJxxxPSZ=nnnn
GJXARENT=nnnn
```

The default page sizes are based on 3390 disk drives. Choose a page size appropriate for your DASD.

If the XA program reentrant pool is large enough, programs are loaded once per CV cycle. If contention occurs in the XA program pools, to reduce runtime I/O you can

increase the default page size for the dictionary load (DDLDCLOD) area from the default of 4276.

You may also want to consider startup resources. Buffer space is defined in the DMCL; the larger the database page sizes, the larger the buffer size. Depending on the:

- Number of buffers
- Size of the database pages
- Number of pages allocated in each buffer pool

it is possible to create a DMCL that has millions of bytes in the buffer pool. The larger the buffer pool, the more startup resources CV requires.

5.2.21 Unicenter TNG Framework for OS/390

`GJSTEPL1=dsname`

When the GJSTEPL1 variable is coded, a user-defined library is added to the appropriate steps in the generated installation JCL.

Ensure one of the following conditions is met prior to starting the CA-IDMS installation process:

- The Unicenter TNG Framework for OS/390 load library is in your OS/390 system LNKLST concatenation.
- The Unicenter TNG Framework for OS/390 load library is in the STEPLIB concatenation in all database jobs of the CA-IDMS installation process. The Unicenter TNG Framework for OS/390 load library must be placed last in the concatenation.

5.3 CA-IDMS Tools Variables

5.3.1 Upgrade

```
WIUPGRAD=YES
         NO ←
```

An Upgrade install is available for CA-IDMS Tools Release 12.0 and later, and:

- Provides enhanced CA-IDMS Tools software for your existing system
- May update database areas, but does not allocate or format database areas

The default is NO, indicating a complete base install. For more information about complete base installs, see 1.3.1, “Complete Base Installation” on page 1-7

If you choose an Upgrade install, (YES):

- CAISAG ignores all parameters for database file allocation
- Specify the name of the existing global DMCL in the GJGLDMCL (5.1.3, “Global DMCL” on page 5-4) variable in VARBLIST
- Additional products **cannot** be installed

5.3.2 Create a CA-IDMS Tools Dictionary

```
WITDMNEW = YES ←
           NO
```

If WITDMNEW=YES, a new CA-IDMS Tools dictionary is allocated using parameters specified in the VARBLIST. If WITDMNEW=NO, the existing CA-IDMS Tools dictionary is specified using WITDMNAM variable.

If this is an Upgrade or Add-On installation, the CA-IDMS Tools dictionary exists. Specify WITDMNEW=NO and specify the name of the dictionary in the WITDMNAM variable.

5.4 CA-Endevor/DB Variables

5.4.1 Existing CA-Endevor/DB into a New SMP/E Environment

```
REINSTALL= YES
           NO ←
```

The REINSTALL variable allows you to install the same version of CA-Endevor/DB into the new CA-IDMS CSI after installing a new release of CA-IDMS.

5.4.2 Upgrade CA-Endevor/DB

```
ABUPGRAD = YES
           NO ←
```

The ABUPGRAD variable is used for specifying an Upgrade install. YES is specified when you have installed an earlier release of CA-Endevor/DB, have existing CCDBs and do not want to allocate new database files.

Note: If CA-Endevor/DB has been installed into the SMP/E environment, you cannot execute an Upgrade install.

5.4.3 CA-Endevor/DB Message Library

```
ABOLDMSG = YES ←
         = NO
ABMSGDSN = msg.libname
```

The ABOLDMSG variable specifies if Endevor for OS/390 is currently installed. ABMSGDSN specifies the name of the message library.

If Endevor for OS/390 is not installed (ABOLDMSG=NO):

- The message library dataset is allocated during the CA-Endevor/DB installation
- Messages are added to the new library.

When ABOLDMSG=YES, Endevor for OS/390 and its message library are currently installed. The CA-Endevor/DB install adds new members to the existing message library.

Chapter 6. CA-IDMS Installation Jobs

6.1 Create SMP/E Environment 6-4
6.1.1.1 Restart Information 6-4
6.2 Download Modules 6-5
6.2.1.1 CAIIPDS Return Codes 6-5
6.2.1.2 Restart Information 6-5
6.3 Create Customized Source 6-6
6.3.1 IDMSOPTI 6-6
6.3.1.1 Restart Information 6-7
6.4 SMP/E RECEIVE Processing 6-8
6.4.1.1 Restart Information 6-8
6.5 SMP/E APPLY Processing 6-9
6.5.1.1 Restart Information 6-9
6.6 SMP/E ACCEPT Processing 6-10
6.6.1.1 Restart Information 6-10
6.7 Miscellaneous SMP/E Processing 6-11
6.7.1.1 Restart Information 6-11
6.8 Backup SMP/E Environment 6-12
6.8.1.1 Restart Information 6-12
6.9 Create USERMOD for WTOEXIT 6-13
6.9.1.1 Restart Information 6-13
6.10 Load the SVC Using CAIRIM 6-14
6.10.1.1 Return Codes 6-15
6.10.1.2 Restart Information 6-15
6.11 Allocate Database Files 6-16
6.11.1.1 Restart Information 6-16
6.12 Create SYSTEM and SYSDIRL Dictionaries 6-17
6.12.1.1 Return Codes: 6-17
6.12.1.2 Restart Information 6-17
6.13 Create APPLDICT and ASF Dictionaries 6-18
6.13.1.1 Restart Information 6-18
6.14 Build Non-SQL Commonweather Database 6-19
6.14.1.1 Restart Information 6-19
6.15 Create SQL Demo Database 6-20
6.15.1.1 Restart Information 6-20
6.16 Generate SYSTEM90 and Format Journals 6-21
6.16.1.1 Restart Information 6-21
6.17 Create UCFCICS Load Module 6-22
6.17.1.1 Restart Information 6-22
6.18 Backup Database Files 6-23
6.18.1.1 Restart Information 6-23
6.19 Populate SAMPJCL 6-24
6.19.1.1 Restart Information 6-24

This chapter describes the jobs executed during the CA-IDMS installation procedure.

The installation jobs are created by the CAISAG utility. Prior to initiating the installation process, review the JCL for each job. Each job consists of one or more job steps; verify the return codes from each step before submitting the next installation job. The installation jobs must be run in order; failure to do so may cause errors.

6.1 Create SMP/E Environment

This job (Job 1) creates the SMP/E environment and allocates these libraries and datasets:

- SMP/E target libraries
- SMP/E distribution libraries
- PPOPTION library used as SMP/E input during APPLY and ACCEPT processing
- SMP/E CSI and SMPxxx datasets
- Staging datasets
  - Indirect libraries used as SMP/E input during APPLY and ACCEPT processing
  - SYSDIRL dataset

This job initializes the CSI and defines the three CSI zones:

- GLOBAL
- IDMSTGT
- IDMSDLB

Note: If you are using JES3, JOB1 must be divided into two jobs because the CSI (VSAM dataset) is defined and initialized in this job.

6.1.1.1 Restart Information

This job can be restarted at any step.

6.2 Download Modules

This job (Job 2) downloads modules from the install tape into the indirect libraries; these are staging libraries and are input to SMP/E processing. After the installation is complete, these libraries can be deleted since they are only required during the installation process.

6.2.1.1 CAIIPDS Return Codes

- **0** — Normal completion.
- **8** — Error messages have been issued. Review the error message text in the output listing to determine the error.
- **12** — This **always** indicates a password error. Review the password(s) supplied by Computer Associates for the product(s) you are installing. Any of the following may be the cause:
  - a wrong password
  - a missing password
  - typing the letter *O* instead of the number *0*
  - typing the letter *I* or *L* instead of the number *1*
  - accidentally over-typing a product code

  Note: Product passwords are always numeric.
- **16** — A parameter error. Contact Computer Associates technical support for assistance.
- **20** — An internal tape format error. Contact Computer Associates technical support for assistance.
- **3x** — An I/O error on SYSIPT.
- **4x** — An I/O error on SYSLST.
- **5x** — An I/O error on SYSPCH.
- **6x** — An I/O error on SYS001.

6.2.1.2 Restart Information

This job can be rerun.

6.3 Create Customized Source

Source modules requiring site-specific parameter values are assembled during the installation process. This job (Job 3) generates the assembler source (macro parms), and copies it into the PPOPTION library using your specified values and CA-IDMS product profile. These source members are assembled and linked by SMP/E as part of the installation process during Job 5 (6.5, “SMP/E APPLY Processing” on page 6-9). Additional customization can be performed before submitting this job. After the install is completed, changes to the source require a user modification (USERMOD).

6.3.1 IDMSOPTI

IDMSOPTI is one of the macro parms generated and copied to the PPOPTION library. Unlike the majority of the macro parms, IDMSOPTI **is not** generated using the values specified in GJSVCNUM and GJCVNUM. IDMSOPTI's function is the rationale for this exception.

IDMSOPTI can be linked with batch programs requiring non-SQL database services to determine the:

- Database
- SVC
- CV number
- Type of access (local mode or CV)

By stipulating these parameters in SYSIDMS, SYSCTL or the program, you have greater flexibility and maintenance is simplified. Therefore, IDMSOPTI is generated with SVC=0 and CVNUM=0 **regardless** of the values specified in the VARBLIST variables GJSVCNUM and GJCVNUM;

Typically, SVC and CV NUMBER are specified in IDMSOPTI only when runtime choices must be restricted.

Example: The SVC is specified in IDMSOPTI. There is maintenance to the SVC and you want to use a different number for the new SVC. To use the new SVC, all programs using IDMSOPTI must be relinked.

If IDMSOPTI specified:

- SVC=0
- CVNUM=0
- SYSCTL=SYSCTL

The new SVC and CV number could be determined from SYSCTL; no programs would need relinking.

6.3.1.1 Restart Information

This job can be rerun.

6.4 SMP/E RECEIVE Processing

This job (Job 4) performs SMP/E RECEIVE processing. The RECEIVE process copies the Modification Control Statements (MCS), associated with the SYSMODs for CA-IDMS, from the SMPPTFIN dataset and places them in the SMPPTS dataset. The global zone of the CSI is updated with the names of the SYSMODs that are successfully RECEIVEd. This job requires the installation tape.

6.4.1.1 Restart Information

To rerun this job:

1. Comment out all FMIDs that were successfully RECEIVEd.
2. Resolve the problem - usually space.
3. Rerun the job.

Example: CGJF000, CGTF000 and CGYF000 were being RECEIVEd, but only CGTF000 was successful. Resubmit the RECEIVE job, as follows:

```
     CGJF000
/*   CGTF000   Already RECEIVEd */
     CGYF000
```

6.5 SMP/E APPLY Processing

This job (Job 5) performs the SMP/E APPLY processing and is a CPU-intensive job. The APPLY process installs the modules of the selected SYSMODs into the Target libraries. To accomplish this task, SMP/E reads the indirect files and any files created by the RECEIVE job, and executes utilities to populate the Target libraries, including the loadlib used in the startup deck. The Target zone of the CSI is updated with a complete description of the selected SYSMODs.

6.5.1.1 Restart Information

To rerun this job:

1. Scan the output for the word “STATUS”, to determine the status of each SYSMOD.
2. For assistance in determining the cause of the errors, see D.2, “Debugging SMP/E Jobs” on page D-16.
3. Comment out all SYSMODs that were successfully APPLYed (this preserves a record of the SYSMODs installed).

```
   CGJF000
/* CGXF000  Already APPLYed */
   CGYF000
```

4. Rerun the job

6.6 SMP/E ACCEPT Processing

This job (Job 6) performs the SMP/E ACCEPT processing. SMP/E reads the indirect files and any files created by the RECEIVE job, and executes utilities to populate the distribution libraries. The distribution zone of the CSI is updated with a complete description of the selected SYSMODs. The successfully ACCEPTed SYSMODs are deleted from the SMPPTS dataset.

6.6.1.1 Restart Information

To rerun this job:

1. Scan the output for the word “STATUS”, to determine the status of each SYSMOD.
2. For assistance in determining the cause of the errors, see D.2, “Debugging SMP/E Jobs” on page D-16.
3. Comment out all SYSMODs that were successfully ACCEPTed (this preserves a record of the SYSMODs installed).

```
   CGJF000
/* CGXF000    Already ACCEPTed */
   CGYF000
```

4. Rerun the job

Note: This job must be run before applying maintenance or modifications to CA-IDMS.

6.7 Miscellaneous SMP/E Processing

This job (Job 7) lists the SYSMODs that have been installed, adds GENASM subentries to selected macro entries, and turns off the RETRYDDN option.

6.7.1.1 Restart Information

To rerun this job:

1. Use D.2, “Debugging SMP/E Jobs” on page D-16 to determine the cause of the error.
2. Resolve the error.
3. If the GENASM step fails, rerun the job to ensure all the updates are completed.

   Note: Updates that were successful the first time produce a non-zero condition code which can be ignored.
4. Rerun the job.

6.8 Backup SMP/E Environment

This job (Job 8) is optional, as determined by your installation parameters in the VARBLIST member. If selected, this job makes a backup of the SMP/E environment. Ensure the /*SETUP cards are suitable for a backup job.

6.8.1.1 Restart Information

This job can be rerun.

6.9 Create USERMOD for WTOEXIT

This job (Job 09) installs an SMP/E USERMOD to create a new startup module. It is generated when the variable GJWTOEX=YES **or** GJSTUP is not equal to nulls. Before submitting this job, you can change any of the #DCPARM parameters. These are in the RHDCPARM member of the PPOPTION library.

The WTOEXIT is used to submit journal offload and print log jobs.

6.9.1.1 Restart Information

Restart processing depends on where the job failed. If the job failed:

- RECEIVE step
  - Correct the problem
  - Rerun the job
- APPCHECK or APPLY step —
  - Add a REJECT statement to the RECEIVE step. For example:

    ```
    SET BOUNDARY (GLOBAL) OPTIONS (IDMSOPT).
    REJECT SELECT (UMSTART) BYPASS(APPLYCHECK).
    RECEIVE SELECT (UMSTART).
    ```

  - Correct the problem
  - Rerun the job

6.10 Load the SVC Using CAIRIM

Job 10 executes CAIRIM and loads the SVC load module, (IGCxxx) into the OS/390 CSA. The SVC load module must be installed into the operating system using CAIRIM; failure to use CAIRIM results in an error when CA-IDMS is started.

CAIRIM is parameter driven and reads the input parms from the sequential file pointed to by the PARMLIB DD statement. To load an SVC, a CAIRIM input line specifies the:

- Product name (CA-IDMS)
- Product's version (GJF0)
- Product initialization routine (GJF0INIT)
- SVC number

The input line for CA-IDMS Release 15.0 SVC:

```
PRODUCT(CA-IDMS) VERSION(GJF0) INIT(GJF0INIT) PARM(SVC=xxx)
```

These modules must be available when CAIRIM is run:

Module Name	Module Type	Location
CAIRIM	Unicenter TNG Framework for OS/390 Services modules	Unicenter TNG Framework for OS/390 Services Loadlib (CAILIB)1
CAS9SEC	Unicenter TNG Framework for OS/390 Services modules	Unicenter TNG Framework for OS/390 Services Loadlib (CAILIB)1
GJF0INIT	CA-IDMS	CA-IDMS Loadlib (APFLIB)
IGCxxx	CA-IDMS	CA-IDMS Loadlib (APFLIB)
RHDCSSFM	CA-IDMS	CA-IDMS Loadlib (APFLIB)

Note:

1 — Library **must** be APF authorized.

The loadlib containing the Unicenter TNG Framework for OS/390 modules must be concatenated in the linklist (LNKLST) or CAIRIM's STEPLIB statement. The location of the Unicenter TNG Framework for OS/390 modules directly affects the CA-IDMS modules.

If the Unicenter TNG Framework for OS/390 modules are concatenated in the linklist (LNKLST), the CA-IDMS modules can be accessed from a:

- Load library in the LNKLST concatentation

- Load library in CAIRIM's STEPLIB concatenation (authorized or unauthorized)
- Authorized load library specified in the CAIRIM LOADLIB parameter

Otherwise, the Unicenter TNG Framework for OS/390 modules are concatenated in CAIRIM's STEPLIB statement, the CA-IDMS modules must be in an authorized library and the load library is also concatenated in CAIRIM's STEPLIB statement.

CAUTION:
If authorized and unauthorized libraries are concatenated in CAIRIM's STEPLIB, all libraries become unauthorized for the duration of the job step.

This is the syntax for the LOADLIB parameter:

```
PRODUCT(desc) VERSION(vers) LOADLIB(dsn) INIT(name) PARM(parm)
```

The dataset name specified in the LOADLIB parameter must be APF authorized. If the LOADLIB parameter is used, a STEPLIB cannot be used.

For additional information on the operation of CAIRIM, refer to the *Unicenter TNG Framework for OS/390 Administrator Guide*.

6.10.1.1 Return Codes

- **0** — Normal completion.
- **S806** — Indicates the LOADLIB parameter was specified on the CAIRIM input line **and** a STEPLIB was specified.

6.10.1.2 Restart Information

This job can be rerun. If the installed SVC must be replaced (the job has run successfully), change the PARM value from:

```
PARM(SVC=xxx)
```

to

```
PARM (REFRESH(SVC=xxx))
```

This indicates you recognize the SVC has been installed by CAIRIM and you wish to replace it.

Note: Before replacing the SVC, ensure all CA-IDMS central versions using the SVC have been shutdown normally. Any system using the SVC that is not shut down, will be unable to process ERU tasks.

6.11 Allocate Database Files

This job (Job 11) allocates the CA-IDMS Release 15.0 database files.

Note: For Upgrade installs, no database files are allocated.

6.11.1.1 Restart Information

This job cannot be rerun; restart the job at the failing step.

6.12 Create SYSTEM and SYSDIRL Dictionaries

This job (Job 12) builds the CA-IDMS Release 15.0 SYSTEM dictionary, including the catalog component, and the SYSDIRL dictionary. It also creates the global DMCL, the DB name table, and generates System 99 which is subsequently used to generate System 90. The actual steps generated for this job depend upon the product mix selected for installation.

CAUTION:
If you are backing up the CA-IDMS/DC UCF system, ensure the /*SETUP cards are suitable for a backup job.

The SVC number specified in VARBLIST is assembled into RHDCSRTT during Job 5. Job 10 (6.10, "Load the SVC Using CAIRIM" on page 6-14) **must** be successfully run before submitting this job.

6.12.1.1 Return Codes:

- **0** — Normal completion.
- **SFnn** — This indicates the system cannot find SVC *nn* (*nn* is the SVC number in hex).

Example: A job abends with a *SFAD*; indicating the system is trying to use SVC 173 (173 = 10x16 + 13x1).

6.12.1.2 Restart Information

This job can be rerun.

6.13 Create APPLDICT and ASF Dictionaries

This job (Job 13) builds the CA-IDMS 15.0 APPLDICT dictionary. If you are installing CA-ADS ASF Option, this job also builds the ASFDICT dictionary.

Steps that update the APPLDICT dictionary **must be** run against all application dictionaries you have created.

6.13.1.1 Restart Information

This job can be rerun or restarted. This job can be restarted at any step, if the step doesn't require a temporary dataset created earlier.

6.14 Build Non-SQL Commonweather Database

This job (Job 14) creates a second DMCL (EMPDMCL) and builds the Commonweather demonstration database, a Non-SQL version of the Employee Skills Demonstration database.

6.14.1.1 Restart Information

This job cannot be rerun. Restart the job at the failing step, provided it does not use a temporary dataset created earlier.

6.15 Create SQL Demo Database

Job 15 builds the SQL demonstration database. An IEFBR14 step is generated if you are not installing the CA-IDMS/SQL Option.

6.15.1.1 Restart Information

This job can be restarted at any step that failed, provided it does not require a temporary dataset created earlier.

6.16 Generate SYSTEM90 and Format Journals

This job (Job 16) builds the sample DC/UCF system, System 90. System 90 is intended as a test system only, designed to be modified by future CA-IDMS changes. You can copy System 90 to use as the basis of your unique Release 15.0 systems.

This job also formats the journal files. If CA-ADS is installed, the ADSOOPTI module is created in the DBA loadlib.

6.16.1.1 Restart Information

This job can be restarted at any step that failed, provided it does not require a temporary dataset created earlier.

6.17 Create UCFCICS Load Module

This job (Job 17) assembles and links #UCFCICS. Member UCFCICS in SAMPJCL performs the same function.

Note: If you install an APAR or maintenance to #UCFCICS, IDMSCINT or RHDCUCFC, this job or the UCFCICS job should be rerun to incorporate the changes.

6.17.1.1 Restart Information

This job can be rerun.

6.18 Backup Database Files

This job (Job 18) performs a backup of all the installed database files.

The /*SETUP cards should be suitable for a backup tape.

6.18.1.1 Restart Information

This job can be rerun.

6.19 Populate SAMPJCL

This job (Job 19) runs a series of IEBGENER steps to create customized members in the SAMPJCL library.

6.19.1.1 Restart Information

This job can be rerun.

Chapter 7. CA-IDMS Post-Installation Tasks

7.1 Environment Created by Installation 7-4
7.2 OS/390 Environment 7-5
7.3 Prepare TP Access Environment 7-6
7.3.1 VTAM Access 7-6
7.3.2 TSO Access 7-6
7.3.3 CICS Access 7-7
7.3.3.1 CICS Access to Multiple DC/UCF Systems 7-7
7.3.3.2 Multiple CICS Regions 7-8
7.4 Creating an Executable System 7-11
7.5 Verify the System is Installed 7-12
7.5.1 System Startup 7-12
7.5.2 Online Verification 7-12
7.5.3 Batch Verification 7-13
7.6 CAIRIM 7-15
7.7 SVC 7-16
7.8 Optional APARs 7-17
7.9 USERMODs 7-18
7.10 Conversion of a 10.2 System 7-19
7.11 Security 7-20

This chapter describes tasks you should complete after installing CA-IDMS.

7.1 Environment Created by Installation

The installation process creates a Release 15.0 environment. It includes:

- Dictionary environment -- A system and application dictionary environment are installed.
- Systems:
  - System 90 -- A sample CA-IDMS Release 15.0 DC/UCF system that can be copied to your DC system number.
  - System 99 -- A non-executable system containing CA-IDMS programs and tasks that can be copied to the new systems you define and generate.
- Four 15.0 load libraries are created:

Load Library	Contains
LOADLIB	■ CA-IDMS system software load modules[1]
DBA.LOADLIB	■ DMCL module -- A 15.0 DMCL is installed into the DBA loadlib and describes the installed 15.0 runtime environment. ■ Database name table -- A database name table that defines the databases and data dictionaries accessible by the R150DMCL is installed in the DBA loadlib. **Note:** The module names were specified in the VARBLIST.
APFLIB	■ SVC ■ Related modules
INTBLOAD	■ Streamlined batch interface modules

Note: [1] - this library should not be authorized.

7.2 OS/390 Environment

There should be no modules from prior releases of CA-IDMS or CA-DISPATCH in the OS/390 LPA. The SYS1.LPALIB should contain no modules beginning with ADS, IDMS, or RHDC.

7.3 Prepare TP Access Environment

This task involves preparing the TP access method for your CA-IDMS system. The access methods covered are VTAM, TSO, and CICS.

7.3.1 VTAM Access

To prepare your VTAM environment for access to CA-IDMS/DC:

1. Define a VTAM line (VTAMLIN) to the CA-IDMS/DC system. In Job 12, a VTAM line (VTAMLIN) was defined to your CA-IDMS/DC using the APPLICATION ID (APPLID) specified in the CA-IDMS VARBLIST.
2. Define an application ID (APPLID) in VTAM to identify the CA-IDMS/DC system.

   Note: The APPLID defined to VTAM must match the APPLICATION ID parameter of the VTAMLIN statement in Job 12.

For additional information concerning VTAM access preparation, refer to *CA-IDMS System Generation*.

7.3.2 TSO Access

If you are installing CA-IDMS/UCF and the UCF/TSO interface, INT-TSO, the load module RHDCUCFT is loaded into the Target library during the installation process. RHDCUCFT is required to access DC/UCF through TSO.

To access a DC/UCF system from TSO, use the CLIST in SAMPJCL member UCFTSO to invoke RHDCUCFT. For reference purposes, the UCFTSO member is shown below. Modify it to reflect your dataset naming conventions.

```
PROC 0
CONTROL NOMSG LIST
FREE F(SYSPRINT,SYSOUT,SYSOUD,SYSLST,SYSUDUMP,SYSCTL,DCMSG)
FREE F(SYSJRNL,J1JRNL,J2JRNL,J3JRNL,J4JRNL,CDMSLIB)
CONTROL MSG
ALLOC F(SYSPRINT) DA(*)
ALLOC F(SYSOUT)   DA(*)
ALLOC F(SYSOUD)   DA(*)
ALLOC F(SYSUDUMP) DA(*)
ALLOC F(SYSCTL)   DA('IDMS.TEST.SYSCTL') SHR
ALLOC F(DCMSG)    DA('IDMS.TEST.SYSMSG.DDLDCMSG') SHR
ALLOC F(SYSJRNL)  DUMMY
ALLOC F(J1JRNL) DUMMY
ALLOC F(J2JRNL) DUMMY
ALLOC F(J3JRNL) DUMMY
ALLOC F(J4JRNL) DUMMY
ALLOC F(CDMSLIB)  DA('IDMS.R150.DBA.LOADLIB',-
                     'IDMS.R150.LOADLIB') SHR
CALL 'IDMS.R150.LOADLIB(RHDCUCFT)'
```

Warning: RHDCUCFT cannot be run from an authorized library.

7.3.3 CICS Access

To access a DC/UCF system from CICS, you must install CA-IDMS/CICS Support and CA-IDMS TP Option for CICS.

Once the install steps have executed successfully, you must add the CA-IDMS PPT entries to your CICS system to define the resources required by CA-IDMS to CICS. This is usually done by a CICS systems programmer using the CICS resource definition online (RDO) or the DFHCSDUP utility to update the CSD file.

Sample CSD resources required by CA-IDMS are shown in SAMPJCL member CICSCSD. If you have not installed CA-IDMS/VSAM Transparency, remove the ESVSEXIT and ESVSVSAM programs.

For a complete description of CA-IDMS CICS support, refer to *CA-IDMS System Operations*. For complete instructions on updating the CICS CSD file, refer to the appropriate IBM documentation.

7.3.3.1 CICS Access to Multiple DC/UCF Systems

The installation created the software required to access multiple (up to 10) DC/UCF systems from one CICS region, provided you can determine at runtime which DC/UCF system should be accessed to service each bind or connect. This determination is made in the OPTIXIT or OPTIQXIT respectively. A single interface module (IDMSINTC) is now able to support every CICS system in your environment, since the TPNAME parameter defaults to null; this causes the CICS system id to be retrieved and used as the TPNAME. From the point of view of a person sitting at a terminal, CICS is called the front-end and CA-IDMS is called the back-end.

Program Interface Module: The front-end must know where to send CA-IDMS calls; this is controlled by the IDMSCINT, which is called the program interface module. Each different program interface module used in CICS requires exclusive use of a fullword in the CICS CWA for locating its interface module (IDMSINTC). The offset of this fullword from the beginning of the CWA is called the CWA displacement. If a CICS region requires more than one interface module, that is accesses more than 10 CVs, multiple IDMSCINT modules specifying unique CWA displacements must be assembled and paired with multiple assemblies of IDMSINTC, one IDMSCINT for each IDMSINTC. Each CICS program that makes CA-IDMS calls must be linked with an IDMSCINT module.

Normally all CA-IDMS calls made by a CICS application program are directed to one and only one IDMSINTC, which is used to send the request to one of the 1 to 10 CVs it manages, that is appropriate for handling the request. These CVs can pass the request to another CV through the facilities of DDS, node names and global dbnames. This limitation is at the request unit or connection level. One CICS task can directly access multiple CVs providing the CA-IDMS calls to each CV are relegated to separate programs and CICS LINKs and XCTLs are used to transfer control between programs. Each version of IDMSCINT should be linked with a unique load module

name, and the label on the IDMSCINT macro should be the same as the load module name.

Interface Modules: IDMSINTC is called the interface module. If a CICS region requires access to more than 10 CVs or if it is impossible to determine at runtime where the request should be sent, multiple IDMSINTC modules must be assembled and linked, specifying the same CWA displacement as their associated IDMSCINT. Compiling the CICSOPT macro specifies runtime options for IDMSINTC. CICSOPT has parameters for the SVC, the CV number, the SYSCTL ddname, and the number of additional CVs that are supported. For maximum flexibility, do not specify SVC or CV number.

In a multi-CV environment using SYSCTL, the SYSCTL ddname is used as the basis for predicting the additional SYSCTL ddnames that are generated. Suffixing the ddname with the digits 1 through 9, up to the number specified for MAXCV generates these additional ddnames. If the new ddnames are more than 8 characters, the eighth character of the ddname is overlaid.

IDMSINTC runs resident in CICS and passes control to the back-end through the SVC. The back-end permits access from this CICS system if the front-end (CICS sysid) has been identified in the system table for the back-end, RHDCFSTB.

IDMSINTL is used like IDMSINTC; occasionally, it provides increased performance, but its capabilities are very limited. IDMSINTL cannot be used with the SQL option, and it doesn't support an IDMSOPTI exit; you must build separate IDMSCINL and IDMSINTL pairs for every CV you wish to access. For more information about IDMSINTC and IDMSINTL, see the *CA-IDMS System Operations* manual.

7.3.3.2 Multiple CICS Regions

Multiple DC/UCF systems can be accessed by one CICS region. Multiple CICS regions can also access one DC/UCF system.

System Table for the Back-end: Multiple CICS regions can access a DC/UCF system. Each CICS region must be identified by a unique front-end system identifier in RHDCFSTB. Code a #FESTENT macro for each front-end, specifying the front-end ID in the FESID parameter.

```
FSTB     TITLE 'FRONT END SYSTEM TABLE FOR UCF'
         #MOPT CSECT=RHDCFSTB,ENV=USER,REGS=NO
         EJECT
         COPY  #UCFDS
         EJECT
RHDCFSTB CSECT
* BLKSIZ= DEFINES THE PAKET SIZE USED FOR THE BULK DATA TRANSFER
FESTABLE #FESTDEF CNT=17
         #FESTENT FESID=CICS
         #FESTENT FESID=CICSBULK,BLKSIZ=4096    CICS DTS ACCESS
         #FESTENT FESID=BATCH
         #FESTENT FESID=BATCBULK,BLKSIZ=16384   BATCH DTS ACCESS
         #FESTENT FESID=IDMS-DC
         #FESTENT FESID=DCXXBULK,BLKSIZ=4096    DC TO DC DTS ACCESS
         #FESTENT FESID=TSO
         #FESTENT FESID=TIAM
         #FESTENT FESID=UTM
         #FESTENT FESID=UTMBULK,BLKSIZ=4096     UTM DTS ACCESS
         #FESTENT FESID=CMS,BLKSIZ=8016
         #FESTENT FESID=CMSBULK,BLKSIZ=3920     CMS DTS ACCESS
         #FESTENT FESID=IMS
         END   FESTABLE
```

TPNAME: Each CICS region that accesses a CA-IDMS CV must be identified by a unique 4-character name. This name may have been specified in the TPNAME parameter of IDMSINTC. Alternately, it may have been defaulted to being omitted, which causes the IDMSINTC program to retrieve and use the CICS sysid that is established at CICS system start-up. For each CICS region, the first 4 characters of the FESID parameter on the #FESTENT macro must match the name that is supplied from the IDMSINTC (or IDMSINTL) module. DC/UCF uses this name to identify the CICS under which a task is running. If you want CA-IDMS/Performance Monitor to track statistics and accounting information, the names must be in the form CICx where x is an alphanumeric character. For more information, see the *CA-IDMS System Operations* manual.

Front-end Table: Each CICS region must have its own front-end table to describe to DC/UCF the devices it controls.

The front-end table is created by assembling these macros:

- front-end table macro (#UCFUFT)
- #UCFCICS macro
- one or more #UCFUTD macros
- #UCFDEND macro.

Example:

```
label    #UCFCICS
         #UCFUFT
         #UCFUTD
         .
         .
         #UCFDEND
```

The #UCFUFT macro specifies the front-end system ID which must be unique for each CICS region.

Front-end Table Load Module: A front-end table load module consists:

- front-end table - unique for each CICS region
- DFHEAI - from DISTLOAD
- DFHEAI0 - from DISTLOAD
- RHDCUCFC - from DISTLOAD
- IDMSCINT - unique for each CV
- UCFPRINT - optional

Each front-end table load module should have a unique name, and the label on the #UCFCICS macro is the entry point of the load module. These modules are resident in CICS.

CICS PPT: Note that since the front-end table is unique for each CICS region and IDMSCINT is unique for each CV, a unique front-end table load module is required for each CA-IDMS system and CICS region combination.

Each CICS program properties table (PPT) must be updated to point to its UCFCICS module(s) (front-end tables) and IDMSINTC module(s) (interface modules). These modules must be available in the CICS STEPLIB concatenation or in the CICS loadlib.

7.4 Creating an Executable System

RHDCPARM is assembled with your site-specific DC system number. If this is a Complete Base install, create the new DC system, either by:

- Copying System 90

  or

- Defining a new DC system

Copying System 90: Using the SGENCOPY member in the SAMPJCL library, copy System 90 to your DC system number, make the necessary modifications and generate the new system.

Warning: Do not use 90 as your system number. This system is reserved for use by Computer Associates.

Defining a New System:

1. Specify the values for the SYSTEM statement parameters, for example:
   - REENTRANT POOL
   - SVC
   - SYSCTL
   - MAX ERUS
2. For the CA-IDMS task and program definitions, copy the tasks and programs from System 99

   Warning: Do not alter task or program definitions copied from System 99.
3. Add the task and program definitions for your applications
4. Add the line definitions for your site
5. Generate the new system by submitting an RHDCSGEN job to execute the sysgen compiler

For more information about creating a DC System, refer to the *CA-IDMS System Generation* manual.

7.5 Verify the System is Installed

Before you begin the verification process, all installation jobs must have completed successfully. The verification process consists of three phases:

1. System startup
2. Online verification
3. Batch verification

Each of these processes is discussed below.

7.5.1 System Startup

To start your CA-IDMS DC/UCF system, submit the STARTUP member in the SAMPJCL library. The system is active when the 'Enter Next Task Code' message appears.

Warning: CA-IDMS DC/UCF systems neither require nor support the use of authorized libraries. Any use of authorized libraries with CA-IDMS represents a potentially serious OS/390 integrity exposure.

Note: CA-IDMS DC/UCF systems use the services of the IDMS SVC to automatically enable themselves to run non-swappable in OS/390 regardless of the value specified in either the OS/390 PPT or SYS1.PARMLIB. To prevent CA-IDMS from automatically forcing itself to run non-swappable, code an S in column 24 of the EXEC parm of the proc that starts CA-IDMS; now CA-IDMS runs exactly as defined in the MVS definitions.

7.5.2 Online Verification

The online verification process consists of the following steps:

1. **DCMT verification -** A CLIST included in the installation executes a large percentage of the DCMT DISPLAY xxxx functions. To use this CLIST, at the DC/UCF 'ENTER NEXT TASK CODE' prompt enter:

   ```
   DCUF SET DICTNAME SYSTEM
   CLIST DCMT-DEMO-CLIST
   ```

   This invokes the CLIST and allows you to view the new DCMT DISPLAY output.

2. **IDD verification -** To validate CA-IDD, sign on to the APPLDICT dictionary and try various CA-IDD commands such as, DISPLAY ALL MODULES, DISPLAY ALL USERS, and DISPLAY ALL SCHEMAS.

3. **Online Command Facility (OCF) verification -** The CA-IDMS Online Command Facility (OCF) was new with Release 12.0. This tool, among other things, replaces the Release 10.x DMCL compiler. To verify OCF, enter the following commands:

- DCUF SET DICTNAME SYSTEM
- OCF
- DISPLAY SEGMENT SYSTEM. This displays the definitions of the segment, files and physical areas that comprise the SYSTEM segment.
- DISPLAY DMCL R150DMCL. This displays the installation DMCL created during the installation process.
- DISPLAY DBTABLE R150DBTB. This displays the database name table created during the installation process.

Most other online products are optional. Exercise these additional products (such as CA-ADS, CA-OLQ, CA-IDMS/Performance Monitor), to verify their installation.

7.5.3 Batch Verification

The batch verification process consists of several steps. An important part of this process occurred during the actual installation. Several CA-IDMS tools and utilities are executed during the installation, including:

- ADSOBSYS - An ADSOOPTI module is created in the DBA loadlib by the ADSOBSYS utility to define the $TOOLTCF (Transfer Control Facility) as a valid CA-ADS runtime application.
- ADSOBTAT - The batch CA-ADS application table load utility (ADSOBTAT) is run to define the $TOOLTCF (Transfer Control Facility) as a valid CA-ADS runtime application.
- IDMSBCF - The CA-IDMS Batch Command Facility (IDMSBCF) controls the execution of most of the CA-IDMS utility programs, DBTABLE processing, and SQL processing. The utilities invoked by IDMSBCF during the installation are:
  - BACKUP
  - FORMAT
  - PRINT PAGE
  - PRINT SPACE
  - RESTORE
  - ROLLBACK
  - ROLLFORWARD
- IDMSCHEM - The non-SQL, Commonweather Demonstration Database schema, EMPSCHM version 100, is added to the APPLDICT DDLDML area using the schema compiler.
- IDMSDDDL - Several data dictionary batch jobs are run to load various entity types (i.e., messages, elements, records and modules) into the DDLDML and DDLDCLOD areas.

- IDMSDMLC - Various programs, including EMPLOAD, are processed using the CA-IDMS COBOL precompiler (IDMSDMLC) during the creation of the non-SQL demonstration database.
- IDMSRPTS - The CA-IDMS Schema Reporter program is executed to list various reports for EMPSCHM Version 100 during the installation of the non-SQL demonstration database.
- IDMSUBSC - The subschema used to define the non-SQL demonstration database, EMPSS01, is loaded and generated using the subschema compiler.
- RHDCMAP1 - The batch mapping compiler is used to load the map, EMPMAP, into the APPLDICT DDLDML area.
- RHDCMPUT - The batch mapping utility module is run to do a PROCESS=ALL for map, EMPMAP, in the APPLDICT dictionary.
- RHDCSGEN - The batch system generation compiler is executed to create SYSTEM 90 and SYSTEM 99.

All jobs executed during the installation process are run in local mode. You may wish to test some of these tasks running against your central version once it is established. You can test any other programs not executed during the installation at your convenience.

7.6 CAIRIM

Job 10 ran CAIRIM as a batch job to install the SVC. This makes the SVC resident until the next IPL. Your systems group can enable the SVC to automatically install at each IPL by adding the CA-IDMS CAIRIM product parameter:

```
PRODUCT(CA-IDMS) VERSION(GJF0) INIT(GJF0INIT) PARM(SVC=xxx)
```

to the CARIMPRM member in your system's PARMLIB. The CAIRIM CAS9 proc, described in the *Unicenter TNG Framework for OS/390 Getting Started Guide*, can be run as a started task at each IPL.

Note: If the CA-IDMS APFLIB loadlib is authorized and CAIRIM loads the SVC from the APFLIB at each IPL, you are loading an updated (and possibly untested) SVC with the first IPL after installing maintenance affecting the SVC. To avoid loading an untested SVC:

- Allocate a new loadlib
- Authorize the new loadlib
- Copy the modules from CA-IDMS's APFLIB loadlib into the new loadlib
- Have CAIRIM load the modules from the new loadlib

The new loadlib is unaffected by CA-IDMS maintenance. If any of these modules are affected by maintenance, they can be copied to the new loadlib after testing.

7.7 SVC

The CA-IDMS Release 15.0 SVC load module is downward compatible. If you are using CA-IDMS Release 14.1, 14.0 or Release 12.0 (Genlevel 9707 or later) you can run both releases on the new SVC. For multiple releases to share the same SVC, the CAIRIM input statements must be modified to avoid CA-IDMS CAIRIM errors. Failure to modify the CAIRIM statement results in the following message:

```
CA-IDMS SVC MUST BE INSTALLED BY CAIRIM BEFORE STARTING THE DATABASE
```

Example: The site is running releases 15.0 and 14.1 and uses SVC 173. These are the CAIRIM PARMLIB input statements:

```
PRODUCT(CA-IDMS) VERSION(GJE1) INIT(GJE1INIT) PARM(SVC=173)
PRODUCT(CA-IDMS) VERSION(GJF0) INIT(GJF0INIT) PARM(REFRESH(SVC=173))
```

Explanation. The first statement installs SVC 173 with version code GJE1 (Release 14.1). The second statement inserts version code GJF0 (Release 15.0) into the CPU you run CAIRIM on while 'refreshing' the same SVC load module. This allows releases 14.1 and 15.0 CA-IDMS CVs to use the same SVC.

7.8 Optional APARs

Most optional APARs are enabled by turning on a bit in the options table. Use the OPTTABLE and GJF0OPTN members in SAMPJCL to decide which bits to turn on. Turn on bits in the options table by modifying the source for RHDCOPTF.

The default RHDCOPTF is in PPOPTION and looks like:

```
#DEFOPTF TYPE=GENERATE
END
```

To turn on bits 5 and 81, code:

```
#DEFOPTF OPT00005
#DEFOPTF OPT00081
#DEFOPTF TYPE=GENERATE
END
```

Assemble and link RHDCOPTF by running the UMODOPTF job in SAMPJCL. This job installs the UM#OPTF USERMOD.

Later, you may decide to enable another APAR. This requires:

1. Modify the RHDCOPTF source in PPOPTION to add (or delete) the required line
2. APPLYing the UM#OPTF USERMOD with REDO

   ```
   SET BOUNDARY(IDMSTGT).
   APPLY SELECT(UM#OPTF) REDO.
   ```

   Note: You can use REDO with this USERMOD because it doesn't contain any JCLIN.

7.9 USERMODs

After the install has been tested, you may want to make changes to your software. This can be done by installing SMP/E USERMODs. USERMODs are discussed in Chapter 13, “User Modification Process,” but it is worth making a note about *REDO* here.

If you need to change to a source module that was copied into PPOPTION during Job 3 (for example, RHDCPARM), install a USERMOD using UMOD1 in SAMPJCL as a guide. This change does not require JCLIN and you can use REDO with any USERMODs that do not contain any JCLIN.

If you need to create a new load module (for example, a new SVC), check the SAMPJCL library for sample USERMODS like UMODSVC. USERMOD creating new load modules, require JCLIN. A USERMOD to include an additional object (like WTOEXIT) into an existing load module also requires JCLIN.

CAUTION:
Do not use REDO with any USERMODs containing JCLIN.

7.10 Conversion of a 10.2 System

The CONVR102 member in SAMPJCL creates a DMCL Syntax Converter loadlib. See the *CA-IDMS Conversion Notebook* for further instructions.

Note: If you are converting from a 10.0 or 10.1 environment, you may NOT use the DMCL syntax converter because of internal changes made to the DMCL structure with the 10.2 release.

7.11 Security

If you elect to secure programs that issue database requests, the security feature of CA-IDMS Release 15.0 may require you to explicitly secure Computer Associates-supplied user mode programs. A list of these user mode programs which issue bind run units can be found in the DLODSECR member of the DISTSRC library. Use this list to create syntax for program security definitions.

Chapter 8. CA-IDMS Tools Installation Jobs

8.1 Update the CA-IDMS SMP/E Environment 8-4
8.1.1.1 Restart Information 8-4
8.2 Download Modules from Tape 8-5
8.2.1.1 CAIIPDS Return Codes 8-5
8.2.1.2 Restart Information 8-5
8.3 Customize CA-IDMS Tools Runtime Options 8-6
8.3.1.1 Restart Information 8-6
8.3.2 CA-IDMS/DML Online Customization 8-7
8.3.3 CA-IDMS/DC SORT Considerations 8-7
8.3.3.1 Examples of CA-IDMS/DC SORT Customization 8-8
8.4 SVC for CA-IDMS Tools 8-11
8.4.1.1 Restart Information 8-11
8.5 SMP/E RECEIVE 8-12
8.5.1.1 Restart Information 8-12
8.6 SMP/E APPLY Processing 8-13
8.6.1.1 Restart Information 8-13
8.7 SMP/E ACCEPT Processing 8-14
8.7.1.1 Restart Information 8-14
8.8 Miscellaneous SMP/E Processing 8-15
8.8.1.1 Restart Information 8-15
8.9 Product-Specific Database Installation Tasks 8-16
8.9.1.1 Return Codes 8-16
8.9.1.2 Restart Information 8-16
8.10 CA-IDMS Tools User Exits 8-17
8.10.1.1 Restart Information 8-17
8.10.2 CA-IDMS Tools User Exits 8-17
8.10.2.1 Determining Required SAMPJCL Members 8-20

This chapter describes the jobs executed during the CA-IDMS Tools installation procedure.

The installation JCL is produced by the CAISAG utility. Always review the installation JCL before submitting the jobs. Each job consists of one or more job steps; verify the return code from each step before submitting the next installation job.

8.1 Update the CA-IDMS SMP/E Environment

This job (Job 1) allocates these libraries and datasets:

- SMP/E Target libraries
  - CA-IDMS/DC SORT CICS load library
  - CA-IDMS/Enforcer load library
  - CWIF0LLT load library
- SMP/E Distribution libraries
- Updates the CA-IDMS CSI Global, Target, and Distribution zones
- CA-IDMS Tools dictionary
- Product database files
- Staging libraries
  - Indirect libraries
  - SASO SPG text dataset

8.1.1.1 Restart Information

This job can be restarted at the failing step.

8.2 Download Modules from Tape

This job (Job 2) downloads modules from the installation tape into the indirect libraries; these are staging libraries and are input to SMP/E processing. After the installation process is complete, these libraries can be deleted since they are only required during the installation process. The CA-IDMS/SASO SPG text is also downloaded from the installation tape.

8.2.1.1 CAIIPDS Return Codes

- **0** — Normal completion.
- **8** — Error messages have been issued. Review the error message text in the output listing to determine the error.
- **12** — This **always** indicates a password error. Review the product passwords supplied by Computer Associates for the products that you are installing. Any of the following may be the cause:
  - a wrong password
  - a missing password
  - typing the letter *O* instead of the number *0*
  - typing the letter *I* or *L* instead of the number *1*
  - accidentally over-typing a product code
- **16** — A parameter error. Contact Computer Associates Technical Support for assistance.
- **20** — An internal tape format error. Contact Computer Associates Technical Support for assistance.
- **3x** — An I/O error on SYSIPT.
- **4x** — An I/O error on SYSLST.
- **5x** — An I/O error on SYSPCH.
- **6x** — An I/O error on SYS001.

8.2.1.2 Restart Information

This job can be rerun.

8.3 Customize CA-IDMS Tools Runtime Options

Certain CA-IDMS Tools incorporate customization modules distributed with default options. This job (Job 3) copies product customization modules into the PPOPTION source library. These modules include changes specified in the VARBLIST variables.

To change runtime options after the install is complete:

1. Change the parameters in the PPOPTION library members shown below.
2. Use JCL found in SAMPJCL member UMOD1 to process the changes as a USERMOD SYSMOD. See Chapter 13, “User Modification Process” for detailed information on building USERMOD SYSMODs.

Product	PPOPTION Library Member	FMID
CA-IDMS/ADS Alive	USGTPARM	CO5F000
CA-IDMS/Database Extractor	USVTPARM	COTF000
CA-IDMS/Dictionary Migrator	USMTPARM	CMVF000
DICTIONARY MIGRATOR ASSISTANT	XDMTPARM	CMUF000
CA-IDMS/Dictionary Module Editor	USETPARM	CM3F000
CA-IDMS/Dictionary Query Facility	DADTPARM	CM4F000
CA-IDMS/DML Online	USDTPARM	CKJF000
CA-IDMS/Enforcer	ESXTPARM	CO3F000
CA-IDMS/MASTERKEY	SSKTPARM	CF8F000
CA-IDMS/ONLINE LOG DISPLAY	USKTPARM	CF9F000
CA-IDMS/SASO	ESSTPARM	CO4F000
GENERAL SORT	TPSPARM	CKPF000

The runtime options are described in Appendix G, “CA-IDMS Tools Runtime Options.”

8.3.1.1 Restart Information

This job can be rerun.

8.3.2 CA-IDMS/DML Online Customization

There are six COPY statements in customization module USDTPARM that follow the specification of parameters for USDCPARM. Each of these COPY statements refers to a source module allowing you to further tailor CA-IDMS/DML Online for your site.

These modules and the session characteristics they control are:

- **USD@DSPC** — Indicates which characters are considered displayable on your terminal devices; that is, any characters not specified here result in an 'INVALID DATA' condition.
- **USD@SSEX** — Specifies which subschemas are excluded from access by CA-IDMS/DML Online. This exclusion is unconditional, and independent of any other security constraints.
- **USD@KYWD** — Defines standard abbreviations recognized by CA-IDMS/DML Online.
- **USD@MOPS** — Redefines the DML command codes recognized by the Menu/Assist mode of CA-IDMS/DML Online.
- **USD@MSTL** — Reformats the static area of the Menu/Assist Mode screen.
- **USD@MTXT** — Specifies the instructional text appearing in the data area of the Menu/Assist Mode screen when this mode is first specified for the session.

Customize these members in the PPOPTION library before running the APPLY job. After completion of the initial tool installation process, these modules reside in the distribution MACLIB (CWIF0MLD).

The beginning of each module contains complete instructions for customization.

8.3.3 CA-IDMS/DC SORT Considerations

During installation, the MAIN and AUX parameters are each assigned a value of 10000 bytes, unless the default values in the TPSPARM member associated with GENERAL SORT are modified.

During each sort session in an application, CA-IDMS/DC SORT acquires main and auxiliary storage as needed, up to the value assigned. (A session is defined by the session number in a SETSORT statement.) For the most efficient sorts, consider the following points:

The **most efficient** sort is one where

- There are many small records in a buffer
- All the buffers reside in main storage

To **increase efficiency** in a given sort session, use a work record containing only the fields needed for sorting. This ensures the work record is as small as possible to meet the requirements.

In an ideal situation,

- Main storage is slightly larger than the space needed for an average sort
- Auxiliary storage adds extra space needed for large sorts

Increasing the proportion of auxiliary storage to main storage may affect response time.

8.3.3.1 Examples of CA-IDMS/DC SORT Customization

At execution time, CA-IDMS/DC SORT allocates sort buffers in multiples of 2000 bytes. To determine the size of a sort buffer:

1. Multiply MINRBUF * Record Length.
2. Round the result up to the next multiple of 2000 bytes.
3. Add 12 bytes for CA-IDMS/DC SORT overhead.

Maximum: Sort buffer size can be no greater that 32K.

Note: CA-IDMS/DC SORT will not split a buffer between main and auxiliary storage. Therefore it is necessary to make efficient use of main and auxiliary storage.

The product of the MINRBUF value and the record length cannot exceed the MAIN value or the AUX value, because there would be insufficient space to store one sort buffer.

In the following four examples, The MAIN and AUX parameters are not changed. The default for each is 10000 bytes.

Example 1:

```
MINRBUF=20
Record Length=100
```

The sort buffer used by CA-IDMS/DC SORT is 2012 bytes:

```
20 * 100 = 2000
2000 is a multiple of 2000
2000 + 12 = 2012
```

CA-IDMS/DC SORT can store four sort buffers (80 records) in main storage and four sort buffers (80 records) in auxiliary storage.

Example 2:

```
MINRBUF=20
Record Length=150
```

The sort buffer used by CA-IDMS/DC SORT is 4012 bytes:

```
20 * 150 = 3000
The next multiple of 2000 is 4000
4000 + 12 = 4012
```

CA-IDMS/DC SORT can store two sort buffers (40 records) in main storage and two sort buffers (40 records) in auxiliary storage.

Example 3:

```
MINRBUF=100  (default)
Record Length=31
```

The sort buffer used by CA-IDMS/DC SORT is 4012 bytes:

```
31 * 100 = 3100
The next multiple of 2000 is 4000
The sort buffer is 4012
```

CA-IDMS/DC SORT can store two sort buffers (200 records) in main storage and two sort buffers (200 records) in auxiliary storage.

Example 4:

```
MINRBUF=100  (default)
Record Length=51
```

The sort buffer used by CA-IDMS/DC SORT is 6012 bytes:

```
51 * 100 = 5100
The next multiple of 2000 is 6000
The sort buffer is 6012
```

CA-IDMS/DC SORT can store one sort buffer (100 records) in main storage and one sort buffer (100 records) in auxiliary storage.

8.4 SVC for CA-IDMS Tools

This job (Job 4) creates customized JCLIN for GSISVCX, the CA-IDMS Tools version of the CA-IDMS SVC exit, IDMSSVCX. GSISVCX gathers billing information for external (batch or CICS) run units by accessing the job accounting field in an OS/390 environment. GSISVCX copies this information into the External Request Element (ERE). If GSISVCX is not invoked by the CA-IDMS SVC, the CA-IDMS/Log Analyzer and CA-IDMS/Task Analyzer Billing Reports and the CA-IDMS/Log Analyzer Billing Record File will not contain valid billing data for external run units. The CA-IDMS/Log Analyzer and CA-IDMS/Task Analyzer Billing Reports and the CA-IDMS/Log Analyzer Billing Record File are dependent upon this ERE data.

Batch Jobs: For batch jobs, GSISVCX copies the job name and the run unit initiation date/time into the ERE (External Request Element) extension.

CICS Tasks: For CICS tasks, GSISVCX accesses the CICS operator-id, terminal-id, and transaction-id, and copies this information into the ERE extension.

Note: Billing information for CA-IDMS run units is always available.

8.4.1.1 Restart Information

This job can be rerun. Ignore warning messages.

8.5 SMP/E RECEIVE

This job (Job 5) performs SMP/E RECEIVE processing. The RECEIVE process copies the Modification Control Statements (MCS), associated with the SYSMODs for CA-IDMS Tools, from the SMPPTFIN dataset and places them in the SMPPTS dataset. The global zone of the CSI is updated with the names of the SYSMODs that are successfully RECEIVEd. This job requires the installation tape.

8.5.1.1 Restart Information

To rerun this job:

1. Comment out all FMIDs that were successfully RECEIVEd.
2. Resolve the problem - usually space.
3. Rerun the job.

Example: CGJF000, CGTF000 and CGYF000 were being RECEIVEd, but only CGTF000 was successful. Resubmit the RECEIVE job, as follows:

```
     CGJF000
/*   CGTF000  */
     CGYF000
```

8.6 SMP/E APPLY Processing

This job (Job 6) performs the SMP/E APPLY processing. The APPLY process installs the modules of the selected SYSMODs into the Target libraries. To accomplish this task, SMP/E reads the indirect files and any files created by the RECEIVE job, and executes utilities to populate the Target libraries, including the loadlib used in the startup deck. The Target zone of the CSI is updated with a complete description of the selected SYSMODs.

8.6.1.1 Restart Information

To rerun this job:

1. Scan the output for the word “STATUS”, to determine the status of each SYSMOD.
2. For assistance in determining the cause of the errors, see D.2, “Debugging SMP/E Jobs” on page D-16.
3. Comment out all SYSMODs that were successfully ACCEPTed (this preserves a record of the SYSMODs installation).

```
   CGJF000
/* CGXF000    Already APPLYED */
   CGYF000
```

4. Rerun the job

8.7 SMP/E ACCEPT Processing

This job (job 7) performs the SMP/E ACCEPT processing. SMP/E reads the indirect files and any files created by the RECEIVE job, and executes utilities to populate the distribution libraries. The distribution zone of the CSI is updated with a complete description of the selected SYSMODs. The successfully ACCEPTed SYSMODS are deleted

8.7.1.1 Restart Information

To rerun this job:

1. Scan the output for the word “STATUS”, to determine the status of each SYSMOD.
2. For assistance in determining the cause of the errors, see D.2, “Debugging SMP/E Jobs” on page D-16.
3. Comment out all SYSMODs that were successfully ACCEPTed (this preserves a record of the SYSMODs installation).

```
   CGJF000
/* CGXF000    Already ACCEPTed */
   CGYF000
```

4. Rerun the job

Note: This job must be run before applying maintenance or modifications to CA-IDMS Tools

8.8 Miscellaneous SMP/E Processing

Job 8 adds SMP/E GENASM entries to the CSI.

8.8.1.1 Restart Information

To rerun this job:

1. Use D.2, “Debugging SMP/E Jobs” on page D-16 to determine the cause of the error.
2. Resolve the error.
3. If the GENASM step fails, rerun the job to ensure all the updates are completed.

   Note: Updates that were successful the first time produce a non-zero condition code which can be ignored.
4. Rerun the job.

8.9 Product-Specific Database Installation Tasks

Job 9 performs multiple product-specific database installation tasks. These tasks are described by comments in the generated JCL.

Note: For an Upgrade or an Add-On install, no steps may be generated for this job, depending on the products being installed.

8.9.1.1 Return Codes

Some tasks may result in a Return Code of 08. Check your output to determine the status of the specific product.

8.9.1.2 Restart Information

This job cannot be rerun.

8.10 CA-IDMS Tools User Exits

Job 10 generates steps to install a customized RHDCUXIT stored in the PPOPTION library. If you have installed a USERMOD for RHDCUXIT, it **must be** removed before running this job.

If you have already added numbered exits to RHDCUXIT for other software products, you must update that version of RHDCUXIT with the required exits for the CA-IDMS Tools.

See the discussion of "Numbered Exits" in the *CA-IDMS System Operations* for a detailed explanation of RHDCUXIT coding conventions.

8.10.1.1 Restart Information

This job can not be rerun or restarted without modification. Modify the COND parameters before rerunning any steps.

8.10.2 CA-IDMS Tools User Exits

User exits are required for these tools:

- CA-IDMS/Master Key
- CA-IDMS/ADS Alive
- CA-IDMS/Task Analyzer

The section describes how to create a customized RHDCUXIT module for the tool mix you are installing. However, the SAMPJCL library downloaded in Step 1 of the installation process contains members that incorporate all the types of changes described below. You simply need to select the correct module names and insert them into JCL supplied in SAMPJCL to generate an appropriate RHDCUXIT. See 8.10.2.1, "Determining Required SAMPJCL Members" on page 8-20 for information on how to select the correct members from SAMPJCL.

Although the specific modifications described below depend on the combination of products being installed, the following is the basic sequence of steps required for user exit definition:

1. Copy RHDCUXIT source from the source library of the installed CA-IDMS system into the PPOPTION source library.

   If you have already added numbered exits to RHDCUXIT for other software products, you must update that version of RHDCUXIT with the required exits for the CA-IDMS Tools.

   Note: The version of RHDCUXIT used by those other software products must be fully compatible with the version of RHDCUXIT distributed with CA-IDMS.

2. Update the copy of RHDCUXIT with appropriate #DEFXIT commands based on the combination of products being installed.

Note: The #DEFXIT entries are positional; that is, #DEFXIT for exit 04 must be the fifth sequential #DEFXIT within RHDCUXIT.

See the discussion of "Numbered Exits" in the *CA-IDMS System Generation* for a detailed explanation of RHDCUXIT coding conventions.

If you added #DEFXIT entries for other products, and any of those exits conflict with the exits required for the CA-IDMS Tools you are installing, consult the installation guides for the other products to determine the method for resolving the problem of multiple users for a given numbered exit.

- **Installing CA-IDMS/ADS Alive**

  These macro statements for the user-invoked exits are required:

```
#DEFXIT ,                                              256
#DEFXIT ,                                              257
#DEFXIT ,                                              258
.....................................................
    ......... repeated for 259 through 330 .........   .
.....................................................
#DEFXIT ,                                              331
#DEFXIT ,                                              332
#DEFXIT MODE=SYSTEM,CALL=DC,EP=USGX333E,AMODE=ANY      333
```

- **Installing CA-IDMS/Master Key**

  When CA-IDMS/Master Key is installed without CA-IDMS/TASK ANALYZER,two system-invoked exits are required:

```
#DEFXIT MODE=SYSTEM,CALL=DC,EP=SSKXT04E,AMODE=ANY   EXIT 04
#DEFXIT MODE=SYSTEM,CALL=DC,EP=SSKXT06E,AMODE=ANY   EXIT 06
```

  These macro statements define the following exits:

Exit Number	Purpose
Exit 4	New Task Exit
Exit 6	Task Termination Exit II

- **Installing CA-IDMS/Task Analyzer**

  When CA-IDMS/Task Analyzer is installed without CA-IDMS/Master Key six system-invoked exits are required:

```
#DEFXIT MODE=SYSTEM,CALL=DC,EP=USFEXT0E,AMODE=ANY   EXIT 00
#DEFXIT MODE=SYSTEM,CALL=DC,EP=USFEXT4E,AMODE=ANY   EXIT 04
#DEFXIT MODE=SYSTEM,CALL=DC,EP=USFEXT5E,AMODE=ANY   EXIT 05
#DEFXIT MODE=SYSTEM,CALL=DC,EP=USFEXTDE,AMODE=ANY   EXIT 13
#DEFXIT MODE=SYSTEM,CALL=DC,EP=USFEXTFE,AMODE=ANY   EXIT 15
#DEFXIT MODE=SYSTEM,CALL=DC,EP=USFEXT3E,AMODE=ANY   EXIT 33
```

These macro statements define the following exits:

Exit Number	Purpose
Exit 0	System Initialization Exit
Exit 4	New Task Exit
Exit 5	Task Termination Exit I
Exit 13	Shutdown Exit
Exit 15	VIB Statistics Exit
Exit 33	Task Analyzer Exit

- **Installing CA-IDMS/Task Analyzer and CA-IDMS/Master Key**

  When both products are installed, seven system-invoked exits are required:

```
#DEFXIT MODE=SYSTEM,CALL=DC,EP=USFEXT0E,AMODE=ANY   EXIT 00
#DEFXIT MODE=SYSTEM,CALL=DC,EP=USFEXT4E,AMODE=ANY   EXIT 04
#DEFXIT MODE=SYSTEM,CALL=DC,EP=USFEXT5E,AMODE=ANY   EXIT 05
#DEFXIT MODE=SYSTEM,CALL=DC,EP=SSKXT06E,AMODE=ANY   EXIT 06
#DEFXIT MODE=SYSTEM,CALL=DC,EP=USFEXTDE,AMODE=ANY   EXIT 13
#DEFXIT MODE=SYSTEM,CALL=DC,EP=USFEXTFE,AMODE=ANY   EXIT 15
#DEFXIT MODE=SYSTEM,CALL=DC,EP=USFEXT3E,AMODE=ANY   EXIT 33
```

  These macro statements define these exits:

Exit Number	Purpose
Exit 0	System Initialization Exit
Exit 4	New Task Exit
Exit 5	Task Termination Exit I
Exit 6	Task Termination Exit II
Exit 13	Shutdown Exit
Exit 15	VIB Statistics Exit
Exit 33	Task Analyzer Exit

  Copy SAMPJCL member USFUEXTX into the PPOPTION source library and update it as follows:

```
          USFUEXTM TYPE=CSECT,                          X
                USREXT0=(NO,,),                         X
                USREXT4=(YES,SSKXT04E,D),               X
                USREXT5=(NO,,),                         X
                USREXTD=(NO,,),                         X
                USREXTF=(NO,,)
```

3. Assemble the updated RHDCUXIT and USFUEXTX. See the table in 8.10.2.1, "Determining Required SAMPJCL Members" for a key to finding the correct sample JCL.
4. Link edit RHDCUXIT as assembled in step 8d with user exit routines appropriate for the tools being installed. See SAMPJCL member TOOLASM for model JCL to perform the assembly and link edit.

Product	Exit Routine
CA-IDMS/Master Key	SSK2IT04 SSK2IT06
CA-IDMS/ADS Alive	USG2333
CA-IDMS/Task Analyzer	USFEXT0 USFEXT4 USFEXT5 USFEXTD USFEXTF USFEXT3 USFEXTW USFUEXT

8.10.2.1 Determining Required SAMPJCL Members

To determine the appropriate SAMPJCL members needed to generate an RHDCUXIT module, use the table below to determine a value for *n* based on your tool mix and substitute that number for *n* in this procedure:

n= **Product.**	**TOOLASMA** **1 2 3**	**TOOLASMB** **4 6**	**TOOLASMC** **5 7**
CA-IDMS/Master Key	Y Y		Y Y
CA-IDMS/ADS Alive	Y Y	Y	Y
	Y Y	Y Y	

1. Select the correct SAMPJCL member to assemble and link RHDCUXIT.
   - *n* = 1 or 2 or 3 - TOOLSASM
   - *n* = 4 or 6 - TOOLSASMB

- *n* = 5 or 7 - TOOLSASMC

2. Edit the select SAMPJCL member by replacing the *n* in TOOLXITn and TOOLLNKn with the value of *n* as shown in the table above.
3. Update the library names and other variables as indicated in the SAMPJCL member you selected from the table above.
4. Submit the job.

Example: You are installing CA-IDMS/Master Key and CA-IDMS/ADS Alive. Follow these steps:

1. Edit TOOLASMA.
2. Replace TOOLXITn with TOOLXIT3.
3. Replace TOOLLNKn with TOOLLNK3.
4. Update library names and other variables as indicated in TOOLASMA.
5. Submit the job.

Note: The destination load library for the new RHDCUXIT must be placed in the startup JCL for your CA-IDMS/DC system such that supercedes any pre-existing versions of RHDCUXIT.

Chapter 9. CA-IDMS Tools Post-Installation Tasks

9.1 Install a Special SVC (SMF Use Only) 9-4
9.2 Convert the CA-IDMS/Database Extractor Database 9-5
9.3 Update the Dictionary 9-6
9.3.1 CA-IDMS/ADS Trace Dictionary Updates 9-6
9.3.2 CA-IDMS/DC SORT Dictionary Updates 9-6
9.3.3 CA-IDMS/DMLO Dictionary Updates 9-6
9.3.4 CA-IDMS/DQF Dictionary Updates 9-7
9.4 Update the CICS Tables 9-8
9.4.1 CA-IDMS/DC SORT 9-8
9.4.2 CA-IDMS/DMLO 9-8
9.5 CA-IDMS/DMLO for TSO 9-9
9.6 Modifying the Sysgen 9-10
9.7 Modify CA-IDMS Startup JCL 9-12
9.8 Cycle CA-IDMS System 9-13
9.9 Install Default JCL 9-14

This chapter describes the post-installation steps in detail. This list summarizes the steps involved in the CA-IDMS Tools post-installation process:

1. Install the SVC for CA-IDMS/Task Analyzer (SMF use only)
2. Convert CA-IDMS\Test Database Builder database
3. Update the dictionary
4. Update the CICS tables for CA-IDMS/DC SORT and CA-IDMS/DML Online
5. Prepare TSO environment for CA-IDMS/DML Online
6. Modify the Sysgen
7. Modify CA-IDMS Startup JCL
8. Cycle CA-IDMS system
9. Install default JCL for CA-IDMS/Database Extractor and CA-IDMS/Dictionary Migrator Assistant

9.1 Install a Special SVC (SMF Use Only)

You may choose to write CA-IDMS/Task Analyzer statistics to an SMF file. To do so, you must install the special type 3 SVC supplied with CA-IDMS/Task Analyzer.

To install the CA-IDMS/Task Analyzer SVC:

1. Modify your SVC table to identify the CA-IDMS/Task Analyzer SVC. This SVC must be identified as a type 3 SVC. No locks are required, and a function code of zero must be specified to allow your CV to use the CA-IDMS/Task Analyzer SVC without OS/390 authorization.
2. Modify the Distribution source library member USFASVCL by providing a value for IGC00*nnn*, where *nnn* is the SVC number you assigned to the CA-IDMS/Task Analyzer SVC in your SVC table. The value must be in signed decimal format (last character is alphabetic). For example, if you assigned 247 as the SVC number in the SVC table, the *nnn* value would be 24G.
3. Link the CA-IDMS/Task Analyzer SVC into your LPA library using the modified Distribution source library member USFASVCL.

9.2 Convert the CA-IDMS/Database Extractor Database

If you are using CA-IDMS/Test Database Builder Release 3.5 or 3.6 and want to preserve the Selection Criteria Specifications for use by CA-IDMS/Database Extractor Release 15.0, you must convert the Release 3.5 or 3.6 database to Release 15.0.

Detailed instructions for converting the database are contained in the "Converting to CA-IDMS/Database Extractor Release 15.0" appendix of the *CA-IDMS/Database Extractor User Guide*.

9.3 Update the Dictionary

The CA-IDMS Tools products are tightly integrated with the data dictionary; this means updating all dictionaries in which you wish to:

- Execute CA-IDMS/ADS Trace
- Execute CA-IDMS/DML Online (CA-IDMS/DMLO) with extended security and/or access restrictions
- Execute CA-IDMS/Dictionary Query Facility (CA-IDMS/DQF)

The default dictionary for each system under which you intend to run CA-IDMS/DC SORT must also be updated.

9.3.1 CA-IDMS/ADS Trace Dictionary Updates

All application dictionaries using CA-IDMS/ADS Trace must be updated with the attributes, records and elements for CA-IDMS/ADS Trace. To update each dictionary:

1. Create an IDMSDDDL job with ATDDDL as input. ATDDDL is located in the CA-IDMS Tools CWIF0SLD library.
2. Execute the job.
3. Review the output and verify the entities were successfully added to the dictionary.

9.3.2 CA-IDMS/DC SORT Dictionary Updates

All application dictionaries using CA-IDMS/DC SORT must be updated with the records and modules for CA-IDMS/DC SORT. To update each dictionary:

1. Create an IDMSDDDL job with TPSDDL as input. TPSDDL is located in the CA-IDMS Tools CWIF0SLD library.
2. Submit the job.
3. Review the output and verify the entities were successfully added to the dictionary.

9.3.3 CA-IDMS/DMLO Dictionary Updates

For a full discussion of updating the dictionary(ies) for CA-IDMS/DMLO, see Appendix H, “CA-IDMS/DMLO Implementations.”

9.3.4 CA-IDMS/DQF Dictionary Updates

CA-IDMS/DQF is a CA-ADS application; the installation process adds the CA-IDMS/DQF application to the CA-IDMS Task Application Table (TAT) in the TOOLDICT dictionary. Each application dictionary using CA-IDMS/DQF must be updated with the CA-IDMS/DQF ADS application.

To update the dictionary:

1. In the CA-IDMS Tools CWIF0SLD library, modify the member DADS120P.
2. Submit DADS120P for execution.
3. Review the output and verify the application was successfully added to the dictionary.

9.4 Update the CICS Tables

To integrate and implement the CA-IDMS Tools from the CICS environment, the associated programs and tasks must be defined to the CICS tables: the PCT (Program Control Table) and the PPT (Processing Program Table).

9.4.1 CA-IDMS/DC SORT

For CA-IDMS/DC SORT:

1. Update the PCT. Using TPSPCT, in the CA-IDMS Tools CWIF0SLD library, add the transaction code entry to the PCT.
2. Update the PPT. Using TPSPPT, in the CA-IDMS Tools CWIF0SLD library, add the program name entry to the PPT.

9.4.2 CA-IDMS/DMLO

For CA-IDMS/DMLO:

1. Update the PCT. Using USDPCT, in the CA-IDMS Tools CWIF0SLD library, add the transaction code entry to the PCT.
2. Update the PPT. Using USDPPT, in the CA-IDMS Tools CWIF0SLD library, add the program name entry to the PPT.

Note: To install CA-IDMS/DMLO on multiple CV's under CICS, see H.2, “Implementing CA-IDMS/DMLO in Multiple CV's Under CICS” on page H-7.

9.5 CA-IDMS/DMLO for TSO

A CLIST can be used to access CA-IDMS/DMLO from your TSO environment.

1. Using USDTSO, in the CWIF0SLD library as a sample, create a CLIST to execute CA-IDMS/DMLO.
2. Modify the CLIST for your site requirements.
3. Add the CA-IDMS loadlibs to the TSO signon STEPLIB.
4. Execute the CLIST.

9.6 Modifying the Sysgen

If you installed CA-IDMS/Log Analyzer or CA-IDMS/Task Analyzer, the sysgen SYSTEM statement must be modified to accommodate:

- Storage requirements for CA-IDMS/Task Analyzer.
- CA-IDMS/Log Analyzer and CA-IDMS/Task Analyzer statistics gathering.

1. Storage Requirements

   a. Before starting CA-IDMS/DC, consider the storage requirements of CA-IDMS/Task Analyzer that can affect the sysgen. These requirements include:

      - Storage pool
      - Program pool
      - Stacksize

      See the "Operations" chapter of the *CA-IDMS/Task Analyzer User Guide*.

      Note: These requirements may be critical to the proper functioning of your environment.

   b. Modify the SYSTEM statement to incorporate the required changes.

2. Statistics Gathering

   CA-IDMS/Log Analyzer and CA-IDMS/Task Analyzer gathers statistical data for all activities except *dialogs* from the "by-task" statistics records. Statistical information for dialogs is gathered from the transaction statistics records. For CA-IDMS to capture this data, the statistics must be requested at the system level.

   a. **By-task Statistics** are controlled through the STATISTICS subparameter of the SYSTEM statement. The minimum specification required is:

```
                          WRITE
STATISTICS TASK { COLLECT }
```

      Note: For CA-IDMS/Log Analyzer, you must specify WRITE; otherwise COLLECT is sufficient.

      Normally, CV writes the by-task statistic records to the log statistics records to the log. With CA-IDMS/Task Analyzer, this action is controlled by the DC STATISTICS option field of the CA-IDMS/Task Analyzer Statistics Plan screen. See the "Operations" chapter of the *CA-IDMS/Task Analyzer User Guide*.

   b. **Dialog Statistics** are specified in the ADSO statement of the sysgen. The DIALOG STATISTICS subparameter of the ADSO statement generates the transaction statistics. The specification required is:

```
                               ALL
DIALOG STATISTICS ON { SELECTED }
```

Note: If you specify SELECTED, CA-IDMS/Log Analyzer, and CA-IDMS/Task Analyzer can only report on dialogs defined with a PROGRAM statement specifying ADSO DIALOG STATISTICS ON.

See the appropriate CA-IDMS guides for complete information on gathering statistics.

9.7 Modify CA-IDMS Startup JCL

To modify the startup JCL for your systems:

- **All CA-IDMS Tools.** Identify the CA-IDMS systems where the online tools are installed. Add the installation libraries to CDMSLIB for the identified CA-IDMS systems.
- **CA-IDMS Tools Database Files.** These CA-IDMS Tools utilize their own database files:
  - CA-IDMS Tools Dictionary
  - CA-IDMS/Database Extractor
  - CA-IDMS/Dictionary Migrator
  - CA-IDMS/DML Online
  - CA-IDMS/Enforcer
  - CA-IDMS/Master Key
  - CA-IDMS/SASO

  which can be added to the CA-IDMS startup JCL, if desired.

  Note: CA-IDMS supports dynamic file allocation; if the file names are specified in the DMCL, they are not required in the startup JCL.
- **CA-IDMS/Enforcer.** Identify the CA-IDMS systems where you want CA-IDMS/Enforcer to run. Add the CA-IDMS/Enforcer load library to the CDMSLIB(s) before the library(ies) containing these CA-IDMS utilities:

IDMSDDDL	RHDCSGEN
IDMSCHDC	IDMSUBSC
IDMSCHEM	IDMSDDDC
RHDCSGDC	IDMSUBDC

  Note: The CA-IDMS/Enforcer load library was allocated and populated during the installation of CA-IDMS Tools.
- **CA-IDMS/DC SORT under CICS.** Add the CA-IDMS/DC SORT CICS load library to the startup JCL for CICS.

9.8 Cycle CA-IDMS System

Cycle your CA-IDMS system.

9.9 Install Default JCL

At the initial installation, you must install the default JCL used by CA-IDMS/Database Extractor and CA-IDMS/Dictionary Migrator Assistant (DMA).

CA-IDMS/Database Extractor JCL is used to execute the batch components of CA-IDMS/Database Extractor. The default CA-IDMS/Database Extractor JCL is contained in SAMPJCL library members:

- USVEXEC - Extracts and loads a database
- USVPSPC - Prints extract specifications
- USVPJCL - Prints extract and load JCL

The DMA JCL and CA-IDMS/Database Extractor JCL allow you to submit jobs to the internal reader from a CA-IDMS system. The DMA JCL is used for the online job submission of CA-IDMS/Dictionary Migrator jobs by DMA. The DMA JCL is any JCL for CA-IDMS/Dictionary Migrator that you are already using or the SAMPJCL library member USMXTRCT. Edit USMXTRCT to remove all parameter statements.

The JCL to upload CA-IDMS/Database Extractor and DMA JCL is included in SAMPJCL library member USVUJCL and XDMBJCL respectively.

Chapter 10. CA-Endevor/DB Installation Jobs

10.1 Update the CA-IDMS SMP/E Environment 10-4
10.1.1.1 Restart Information 10-4
10.2 Download Modules from Tape 10-5
10.2.1.1 CAIIPDS Return Codes 10-5
10.2.1.2 Restart Information 10-5
10.3 SMP/E RECEIVE Processing 10-6
10.3.1.1 Restart Information 10-6
10.4 SMP/E APPLY Processing 10-7
10.4.1.1 Restart Information 10-7
10.5 SMP/E ACCEPT Processing 10-8
10.5.1.1 Restart Information 10-8
10.6 Database Installation Tasks 10-9
10.6.1.1 Restart Information 10-9

This chapter describes the jobs executed during the CA-Endevor/DB installation process.

These installation jobs are created by the CAISAG utility. Prior to initiating the installation procedures, review the JCL for each job. Each job contains one or more job steps; verify the return codes from each step before submitting the next installation job.

10.1 Update the CA-IDMS SMP/E Environment

This job (Job 1) allocates these libraries and datasets:

- SMP/E Target library
- SMP/E Distribution libraries
- Updates the CA-IDMS CSI Global, Target, and Distribution zones
- CA-Endevor/DB database files
- Indirect libraries

10.1.1.1 Restart Information

This job can be restarted at any step.

10.2 Download Modules from Tape

This job (Job 2) downloads modules from the installation tape into the indirect libraries; these are staging libraries and are input to SMP/E processing. After the installation process is complete, these libraries can be deleted since they are only required during the installation process.

10.2.1.1 CAIIPDS Return Codes

- **0** — Normal completion.
- **8** — Error messages have been issued. Review the error message text in the output listing to determine the error.
- **12** — This **always** indicates a password error. Review the product passwords supplied by Computer Associates for the products that you are installing. Any of the following may be the cause:
  - a wrong password
  - a missing password
  - typing the letter *O* instead of the number *0*
  - typing the letter *I* or *L* instead of the number *1*
  - accidentally over-typing a product code
- **16** — A parameter error. Contact Computer Associates Technical Support for assistance.
- **20** — An internal tape format error. Contact Computer Associates Technical Support for assistance.
- **3x** — An I/O error on SYSIPT.
- **4x** — An I/O error on SYSLST.
- **5x** — An I/O error on SYSPCH.
- **6x** — An I/O error on SYS001.

10.2.1.2 Restart Information

This job can be rerun.

10.3 SMP/E RECEIVE Processing

Job 3 performs SMP/E RECEIVE processing. The RECEIVE process copies the Modification Control Statements (MCS), associated with CA-Endevor/DB from the SMPPTFIN dataset and places them in the SMPPTS dataset. The global zone of the CSI is updated with the names of the SYSMODs that have been successfully RECEIVEd.

10.3.1.1 Restart Information

To rerun this job:

1. Comment out all FMIDs that were successfully RECEIVEd.
2. Resolve the problem - usually space.
3. Rerun the job.

Example: CGJF000, CGTF000 and CGYF000 were being RECEIVEd, but only CGTF000 was successful. Resubmit the RECEIVE job, as follows:

```
     CGJF000
/*   CGTF000  */
     CGYF000
```

10.4 SMP/E APPLY Processing

This job (Job 4) performs the SMP/E APPLY processing. The APPLY process installs the modules into the Target library. The Target zone of the CSI is updated with a complete description of the modules. To accomplish this task, SMP/E reads the indirect files and any files created by the RECEIVE job and executes utilities to populate the Target library.

10.4.1.1 Restart Information

To rerun this job:

1. Scan the output for the word “STATUS”, to determine which the status of each SYSMOD.
2. For assistance in determining the cause of the errors, see D.2, “Debugging SMP/E Jobs” on page D-16.
3. Comment out all SYSMODs that were successfully APPLYed (this preserves a record of the SYSMODs installation).

```
   CGJF000
/* CGXF000    Already APPLYed */
   CGYF000
```

4. Rerun the job

10.5 SMP/E ACCEPT Processing

This job (Job 5) performs the SMP/E ACCEPT processing. SMP/E reads the indirect files and any files created by the RECEIVE job, and executes utilities to populate the distribution libraries. The distribution zone of the CSI is updated with a complete description of the selected SYSMODs. The successfully ACCEPTed SYSMODs are deleted from the SMPPTS dataset.

10.5.1.1 Restart Information

To rerun this job:

1. Scan the output for the word “STATUS”, to determine which the status of each SYSMOD.
2. For assistance in determining the cause of the errors, see D.2, “Debugging SMP/E Jobs” on page D-16.
3. Comment out all SYSMODs that were successfully ACCEPTed (this preserves a record of the SYSMODs installation).

```
   CGJF000
/* CGXF000  Already ACCEPTed */
   CGYF000
```

4. Rerun the job

Note: This job must be run before applying maintenance or modifications to CA-Endevor/DB.

10.6 Database Installation Tasks

Job 6 performs database installation tasks.

Warning: Before running this job, backup your SYSTEM and SYSDIRL segments.

10.6.1.1 Restart Information

1. Restore the SYSTEM and SYSDIRL files from backup.
2. Rerun the job.

Chapter 11. CA-Endevor/DB Post Installation Tasks

11.1 Modify CA-IDMS Startup JCL 11-3
11.2 Cycle CA-IDMS 11-4

11.1 Modify CA-IDMS Startup JCL

To modify the start-up JCL for your CA-IDMS system, add the CA-Endevor/DB target loadlib (CABF0LLT) to CDMSLIB for your CA-IDMS system.

11.2 Cycle CA-IDMS

Cycle your CA-IDMS system.

Chapter 12. Maintenance Process

12.1 Test Fixes and APARs 12-4
12.1.1.1 Installing APARs or Test Fixes: 12-4
12.1.1.2 Removing APARS and Test Fixes 12-5
12.1.1.3 Test Fixes and Corresponding APARs 12-5
12.1.1.4 ACCEPTing Test Fixes, Optional APARs and Published APARS 12-6
12.1.2 APAR tapes 12-6
12.1.3 Informational APAR, or PML (Product Maintenance Letter) 12-6
12.2 Maintenance Tapes 12-7
12.3 Installing Maintenance Tapes 12-8
12.3.1 Review the Cover Letter 12-8
12.3.2 Download Maintenance SAMPJCL Library 12-8
12.3.3 Review the #README Member 12-9
12.3.4 Customize the JOBCARD Member 12-9
12.3.5 Customize the SMP/E Procedure 12-9
12.3.6 Remove USERMODS 12-9
12.3.7 RECEIVE Maintenance 12-10
12.3.8 Run Genlevel-specific Jobs 12-10
12.3.9 APPLY CHECK Maintenance 12-10
12.3.10 RESTORE Applicable SYSMODs 12-10
12.3.11 APPLY Maintenance 12-10
12.3.11.1 Non-zero Condition Codes 12-11
12.3.12 ACCEPT CHECK Maintenance 12-11
12.3.13 ACCEPT Maintenance 12-11
12.3.13.1 Non-zero Condition Codes 12-11
12.3.14 Backup SMP/E 12-11
12.3.15 Reinstall User Modifications 12-12
12.3.16 Re-APPLY Applicable SYSMODs 12-12
12.3.16.1 Non-zero Condition Codes 12-12
12.3.17 Install Special Processing Maintenance 12-13
12.3.18 Save all Materials and Output 12-13

This chapter describes how to use SMP/E to apply various types of updates to your system.

12.1 Test Fixes and APARs

CA-TCC is the primary vehicle for the delivery of test fixes and APARs. Test fixes are written for clients experiencing problems for which there is no published APAR. Verified fixes become published APARS.

- Test fixes begin with T and are packaged as ++USERMODs
- Published APARs begin with LO and are packaged as ++APARs.

SMP/E treats test fixes and published APARs the same.

Published APARs and test fixes usually update load modules with superzap VER/REP statements. The ++ZAP statement in the APAR/fix names an object module and SMP/E calls superzap to zap every load module containing that object.

Some APARs/fixes update macros with IEBUPDTE statements. SMP/E reassembles and links a source module if it has been associated with the macro by a GENASM entry in the CSI.

12.1.1.1 Installing APARs or Test Fixes:

Installing APARs or test fixes requires these steps:

1. RECEIVE

   ```
   SET BOUNDARY(GLOBAL).
   RECEIVE SELECT(xxxxxxx).
   ```

2. APPLY CHECK

   ```
   SET BOUNDARY(IDMSTGT).
   APPLY CHECK SELECT(xxxxxxx).
   ```

3. APPLY

   ```
   SET BOUNDARY(IDMSTGT).
   APPLY SELECT(xxxxxxx).
   ```

SMP/E does not allow multiple updates (zaps) to the same module in a single APPLY command. To install two APARs for the same module, use separate APPLY commands:

```
SET BOUNDARY(IDMSTGT).
APPLY SELECT(apar1).
APPLY SELECT(apar2).
```

Error Messages:

A RECEIVE command failing with a NO APPLICABLE ++VER message might indicate a typographical error in the FMID, or the wrong CSI name in the JCL.

An APPLY command failing with successful AMA messages at the bottom of the output, indicates one of the following:

- IDRDATA is full
- There is a blank line after the last REP statement

RESTORE the APAR, correct the problem, and reinstall the fix.

12.1.1.2 Removing APARS and Test Fixes

It may be necessary to remove an APAR and test fix from your system. This is a two step process.

1. RESTORE CHECK

   ```
   SET BOUNDARY(IDMSTGT).
   RESTORE CHECK SELECT(xxxxxxx).
   ```

2. RESTORE

   ```
   SET BOUNDARY(IDMSTGT).
   RESTORE SELECT(xxxxxxx).
   ```

The *RESTORE* command usually removes the SYSMOD from both the target and global zones. When the *NOREJECT* option is set in the global zone, the *RESTORE* removes the APAR or test fix from the target zone only; a *REJECT* is required to remove the SYSMOD from the global zone. The CA-IDMS install does not set the *NOREJECT* option.

- REJECT

  ```
  SET BOUNDARY(GLOBAL).
  REJECT SELECT(xxxxxxx).
  ```

CAUTION:
Never use BYPASS(ID) on a *RESTORE* command. It can corrupt the CA-IDMS software. SMP/E returns good condition codes even if the software is corrupted.

12.1.1.3 Test Fixes and Corresponding APARs

A published APAR cannot be installed when its corresponding test fix is already installed. Though the published APAR supersedes the test fix, SMP/E does not reverse the VERs and REPs. If the test fix is installed the published APAR will not verify.

If the test fix is installed and you wish to install the published APAR, there are two options:

- RESTORE the test fix and install the published APAR afterwards.
- Delete the VER statements from the APAR and change the REP statements to VER statements. SMP/E installs the APAR, which verifies it matches the test fix.

12.1.1.4 ACCEPTing Test Fixes, Optional APARs and Published APARS

The guidelines for ACCEPTing the fixes varies depending on the type of update you are installing.

- **Never ACCEPT** a test fix.
- **Never ACCEPT** an *optional* APAR.
- A published APAR can be ACCEPTed under limited conditions; though even if these conditions are met, it is not recommended:
  - Identify the genlevel of the next maintenance tape you are installing
  - Identify the 'high APAR' on the tape
  - The APAR under consideration for ACCEPTing is less than or equal to the high APAR of the next maintenance tape

  Under these conditions, the next maintenance tape supersedes the APAR and can be installed on top of it.

  Warning: If the APAR does not meet these criteria and it is ACCEPTed, you will have trouble installing the next maintenance tape.

12.1.2 APAR tapes

You may be sent a tape containing numerous APARS called an APAR tape. These tapes allow you to install several APARs at one time rather than downloading them individually from TCC. You are not required to install APAR tapes. APARs delivered on an APAR tape are incorporated into the next maintenance tape.

12.1.3 Informational APAR, or PML (Product Maintenance Letter)

The informational APAR or PML is official information relative to a product. PMLs are delivered via hard copy, CA-TCC and on maintenance tapes. The naming convention for PMLs is 'LIxxxxx'. 'LI' identifies an information APAR, followed by an identification number.

12.2 Maintenance Tapes

Maintenance tapes are automatically shipped to every site. They contain the official PTF SYSMODs for the CA-IDMS product line and are formatted for SMP/E processing. Maintenance tapes for CA-IDMS product line are not cumulative; you cannot skip maintenance tapes.

The volume serial number for maintenance tapes follows the format *F0GJsM*, where:

- **F0** represents Release 15.0
- **GJ** is the Computer Associates two-character product ID for CA-IDMS
- **s** is the service pack
- **M** indicates a maintenance tape

Refer to the external label of the tape for the current VOLSER.

The maintenance tape includes:

- A partitioned dataset called SAMPJCL containing the sample JCL and instructions to install the maintenance. The cover letter provides instructions for downloading the SAMPJCL file from tape.

  In the SAMPJCL library, most of the member names begin with four alphanumeric characters designating the product family and the current release. In the remainder of this chapter, they are referred to as **pprr**.

Product Family	Code designator
CA-IDMS	GJF0
CA-IDMS Tools	WIF0
CA-Endevor/DB	ABF0

- CAI.BASE.SAMPJCL, a copy of the SAMPJCL file delivered on the current genlevel of the base tape.
- PTFs are official replacement module or modules, resolving one or more problems. Typically, a PTF is an accumulation of APARs. PTFs are always delivered on maintenance tapes. The CA-IDMS PTF naming convention is *ccnnSPm*, where:
  - **cc** is the component code
  - **nn** is a number used by Computer Associates
  - **m** is the service pack number

12.3 Installing Maintenance Tapes

The jobs required vary from genlevel to genlevel. These are the typical steps for installing a CA-IDMS maintenance tapes:

1. Review the cover letter
2. Download the SAMPJCL library
3. Review the #README member
4. Customize the JOBCARD member
5. Customize the SMP/E procedure
6. Remove USERMODs
7. SMP/E RECEIVE processing
8. Execute genlevel-specific jobs
9. SMP/E APPLY CHECK processing
10. SMP/E RESTORE CHECK and RESTORE processing
11. SMP/E APPLY processing
12. SMP/E ACCEPT CHECK processing
13. SMP/E ACCEPT processing
14. Backup SMP/E environment
15. Reinstall USERMODs
16. Review test fixes for published versions
17. Run genlevel-specific jobs
18. Save all materials and output

12.3.1 Review the Cover Letter

Review the cover letter changes to the installation procedure. The cover letter may contain work-arounds for recently discovered installation problems.

12.3.2 Download Maintenance SAMPJCL Library

The cover letter contains the instructions for downloading the SAMPJCL library. Once this job has ended, your library contains all the JCL needed to perform maintenance for CA-IDMS, CA-IDMS Tools, or CA-Endevor/DB, depending on which SAMPJCL file was downloaded.

12.3.3 Review the #README Member

The #README member describes the jobs required for a particular maintenance genlevel; some genlevels require more jobs than others.

12.3.4 Customize the JOBCARD Member

Customize the JOBCARD member for your site, suitable for a long-running job that does not require tape mounts.

12.3.5 Customize the SMP/E Procedure

If you do not have a cataloged procedure set up for SMP/E jobs, customize member **pprr**SMPE from the SAMPJCL library to use as an in-stream PROC. Supply the name of your CSI.

12.3.6 Remove USERMODS

All USERMODs created with JCLIN statements **must** be removed before installing the maintenance tape. One of two techniques can be used to remove USERMODs:

- SMP/E System Backup - Restore the SMP/E environment from a backup created after accepting the last maintenance tape and prior to installing any USERMODs or APARs.
- SMP/E RESTORE - Create and run a SMP/E RESTORE job to remove all the USERMODs. The job should include a RESTORE CHECK step before the RESTORE step; if the RESTORE CHECK step is unsuccessful, do not execute the RESTORE step.

Example:

```
//RESCHECK EXEC PGM=GIMSMP,PARM='DATE=U'
//SMPCSI   DD DISP=SHR,DSN=IDMS.R150.CSI
//SMPCNTL  DD *
 SET BOUNDARY(IDMSTGT).
 RESTORE CHECK SELECT ( UMODSVC
                        CV1SRTT
                        CV2SRTT
                        UM#WTOX
                        UM#CTAB
                       ).
/*
//RESTORE  EXEC PGM=GIMSMP,PARM='DATE=U',COND=(0,LT)
//SMPCSI   DD DISP=SHR,DSN=IDMS.R150.CSI
//SMPCNTL  DD *
 SET BOUNDARY(IDMSTGT).
 RESTORE       SELECT ( UMODSVC
                        CV1SRTT
                        CV2SRTT
                        UM#WTOX
                        UM#CTAB
                       ).
```

There is a *NOREJECT* option in the CSI. When this option is turned off, (the default) RESTOREing a SYSMOD automatically triggers the SYSMOD to be REJECTed. If *NOREJECT* is turned on, you must explicitly REJECT the SYSMOD.

12.3.7 RECEIVE Maintenance

SAMPJCL member **pprr**MREC automatically RECEIVEs all the PTFs corresponding to the products you have installed. To RECEIVE the PTFs:

1. Modify **pprr**MREC JCL to include the customized JOBCARD and the tailored **pprr**SMPE member (if needed)
2. Submit the job
3. Verify the RECEIVE processing ran successfully

12.3.8 Run Genlevel-specific Jobs

These jobs, if any, are described in the #README member.

12.3.9 APPLY CHECK Maintenance

This step identifies SMP/E SYSMODs that prevent PTF application. To identify the SYSMODS:

1. Modify **pprr**MAPC JCL to include the customized JOBCARD and the tailored **pprr**SMPE member (if needed)
2. Submit the job
3. Execute **pprr**MRES to remove all APARs causing MODID errors
4. Repeat steps 2 and 3 until the APPLY CHECK processing runs successfully

12.3.10 RESTORE Applicable SYSMODs

This step RESTOREs SMP/E SYSMODs (APARs or test fixes) identified by APPLY CHECK processing. If you do not have any SYSMODs to RESTORE, continue to the next step.

12.3.11 APPLY Maintenance

SAMPJCL member **pprr**MAPP APPLYs all PTFs from this maintenance tape. To APPLY maintenance:

1. Modify **pprr**MAPP JCL to include the customized JOBCARD and the tailored **pprr**SMPE member (if needed)
2. Submit the job
3. Verify the APPLY processing ran successfully

12.3.11.1 Non-zero Condition Codes

If the return code:

- **= 0** Go to 12.3.12, “ACCEPT CHECK Maintenance”
- **= 4** Review the reports to ascertain if a problem exists (GIM65901W messages cannot be ignored)
- **> 4** Go to D.2, “Debugging SMP/E Jobs” on page D-16

12.3.12 ACCEPT CHECK Maintenance

To ACCEPT CHECK the new maintenance:

1. Modify **pprr**MACH JCL to include the customized JOBCARD and the tailored **pprr**SMPE member (if needed)
2. Submit the job
3. Verify the ACCEPT CHECK processing ran successfully

12.3.13 ACCEPT Maintenance

Maintenance JCL member **pprr**MACC ACCEPTs all PTFs. To ACCEPT the new maintenance:

1. Modify **pprr**MACC JCL to include the customized JOBCARD and the tailored **pprr**SMPE member (if needed)
2. Submit the job
3. Verify the ACCEPT processing ran successfully

12.3.13.1 Non-zero Condition Codes

If the SMP/E ACCEPT completes with a return code greater than 4:

1. **Review the output carefully before continuing**
2. Correct the problem
3. Submit the job, without changes

12.3.14 Backup SMP/E

The entire SMP/E environment should be backed up before installing any user modifications, optional APARs or test fixes. The SMPPTS, SMPLTS, SMPMTS, SMPSTS and SMPSCDS datasets should be empty and no SYSMODs should be in error status. When these conditions are met, the SMP/E system is stable. An unstable system may harbor a bug which is not discovered until after all the stable backups have aged and been discarded.

At a minimum, the backup must contain the following datasets because these datasets must be synchronized with each other.

- CSI
- SMPPTS
- SMPLTS
- SMPMTS
- SMPSTS
- SMPSCDS
- Target Loadlibs
- Distribution Libraries

The following datasets are optional to include in the backup:

- DBA.LOADLIB
- PPOPTION
- SMPLOG
- SMPLOGA

12.3.15 Reinstall User Modifications

Reinstall any site-specific user modifications (USERMODs) containing JCLIN.

12.3.16 Re-APPLY Applicable SYSMODs

Determine if any APARs and test fixes that were removed (RESTOREd) because of MODID errors have been sourced, or if optional APARs have been written to replace them. To reinstall APARs or test fixes:

1. Update the PRE parm on the ++VER statement to account for the new maintenance
2. Modify **pprr**MRAP JCL to include the customized JOBCARD and the tailored **pprr**SMPE member (if needed)
3. Submit the job
4. Verify the APPLY processing ran successfully

12.3.16.1 Non-zero Condition Codes

If the SMP/E APPLY completes with a return code greater than 4, see D.2, “Debugging SMP/E Jobs” on page D-16 for help in resolving the problem.

12.3.17 Install Special Processing Maintenance

The CA-IDMS maintenance tape may contain changes to CA-IDMS elements which require special processing, such as:

- Dictionary entities
- Messages
- Reports
- CICS PPT entries
- SVC updates

Installing maintenance to these elements may require you to perform one or more of these tasks:

- Running IDMSDDDL
- Running RHDCSGEN
- Updating your CICS PPT entries
- Running CAIRIM to dynamically refresh the CA-IDMS SVC

If the SVC is modified by a maintenance tape, the cover letter states the SVC must be relinked. If SMP/E linked the SVC load module, it automatically relinks the SVC when maintenance is applied. If the SVC load module is created by a USERMOD, it is relinked when the USERMOD is applied.

Maintenance sample JCL member **pprr**SPEC contains control statements for most required additional maintenance processing. To apply maintenance requiring special processing for the elements listed above, or additional processing to non-SMP/E-processed Target load libraries: perform these steps:

1. Edit the JCL as necessary
2. Submit the job
3. Verify all steps ran successfully and completed with a return code equal to or lower than that specified on the EXEC statement of the JCL

Important Refer to the maintenance tape cover letter for additional steps which may be required to activate the maintenance.

12.3.18 Save all Materials and Output

Save all of the maintenance materials and output from the maintenance process. This material is essential for timely and accurate Computer Associates maintenance and support of the product.

Chapter 13. User Modification Process

13.1 Overview of User Modifications 13-4
13.2 USERMOD Syntax 13-5
13.3 Required Statements 13-6
13.3.1 ++USERMOD 13-6
13.3.2 ++VER 13-6
13.4 Element Replacement 13-7
13.5 Element Updates 13-8
13.6 JCLIN 13-9
13.7 JCL to Install USERMODs 13-11
13.7.1 RECEIVE Step 13-11
13.7.2 APPLY Steps 13-11
13.7.2.1 Input Libraries 13-12
13.7.2.2 Output Libraries 13-12
13.8 Installing USERMODs 13-13
13.8.1 MODID 13-13
13.9 REDO 13-14
13.10 Restoring USERMODs 13-15
13.11 SUP Parameter 13-16
13.12 MULTIPLE CVs 13-17
13.13 GIM65901W 13-18
13.14 Multiple Load Module Versions (GIM41101E) 13-19

This chapter describes the procedures to follow when customizing or modifying your system after the initial installation.

13.1 Overview of User Modifications

An installed CA-IDMS system can be modified. All site-specific modifications to your system affecting the runtime libraries can be processed by SMP/E as user modifications (USERMODs); SMP/E maintains a record of the changes. These USERMODs must be removed from your system prior to installing a maintenance tape and reapplied after the maintenance tape is completed.

The instructions for installing a maintenance tape direct you to first remove your USERMODs. This refers to USERMODs constructed as described in this chapter.

A user modification is commonly used for:

- Changing assembler parameters.
- Creating a new load module, e.g., a new startup module.

13.2 USERMOD Syntax

Like an APAR, a USERMOD consists of modification control statements (MCS). MCS statements:

- Begin with two plus signs in column 1.
- End with a period.
- May span more than one line, no continuation characters are required. Two MCS statements:

```
++USERMOD(UMODSVC).
++VER(Z038)
  FMID(CGJF000)
  PRE(GJ09610).
```

- May contain blanks between operands and parameters.
- Parameters may be separated by blanks and/or commas.

```
++VER  (Z038)
  FMID ( CGJF000 )
  PRE  ( GJ09610 GJ09711 , LO12345)
  .
```

- May contain comments between operands, but not between modification control statements.
- Comments begin with /* and end with */.

  Example: Valid Statements:

```
++USERMOD(UMOD001).
++VER(Z038) FMID(CGJF000) /*********************/
                          /*  SUPPLY THE RIGHT */
                          /*  FMID FOR THE     */
                          /*  MODULE.          */
                          /*********************/.
```

  Example: Invalid Statements:

```
++USERMOD(UMOD001).
++VER(Z038) FMID(CGJF000). /*********************/
                           /*  SUPPLY THE RIGHT */
                           /*  FMID FOR THE     */
                           /*  MODULE.          */
                           /*********************/
```

13.3 Required Statements

Every USERMOD begins with the same two statements:

1. ++USERMOD (*usermod-name*).
2. ++VER (Z038) FMID (*fmid*).

13.3.1 ++USERMOD

The first line of a USERMOD begins with a ++USERMOD statement and specifies the name of the USERMOD. USERMOD names must be seven characters long and unique among the other SYSMOD names in the CSI.

Example:

```
++USERMOD(WTOEXIT).
++USERMOD ( UMODSVC ) .
++USERMOD (CV15#00)      /* 1. SVC */ .
```

13.3.2 ++VER

The second MCS statement is the ++VER or environment statement. The ++VER statement always contains Z038 indicating OS/390. The CSI is initialized to Z038 during the base install; the ++VER statement confirms the environment.

Each ++VER statement contains a FMID operand; every element the USERMOD updates or replaces must belong to this FMID.

Example:

```
++VER(Z038) FMID (CGJF000)   /* CA-IDMS/DB   */.
++VER(Z038) FMID (CFIF000)   /* CICS SUPPORT */ .
++VER(Z038) FMID (CGJF005)   /* SVC */.
```

Note: Never use a SUP parameter in your USERMODs.

13.4 Element Replacement

The last statements in a USERMOD indicate which elements (modules) are being replaced, added or updated. The module name is coded in parenthesis. These statements add or replace elements:

```
++SRC(#SVCOPT)  ... .
++MOD(RHDCWTL)  ... .
++MAC(#UCFCICS) ... .
```

SMP/E must know the location of the new or replacement module. The TXLIB operand specifies the ddname of the library (discussed later) containing the source or object modules. The TXLIB parameter is usually PPOPTION.

```
++SRC(UCFCICS) TXLIB(PPOPTION)  /* PPOPTION IS A DDNAME */.
++SRC(RHDCPARM) TXLIB(PPOPTION) /* RHDCPARM IS IN PPOPTION */.
```

Changing a parameter of a module that was assembled and linked during the install, requires modifying the source in the PPOPTION library and installing a USERMOD with a ++SRC statement for the module. This type of change requires three statements.

Example: This USERMOD reassembles and relinks RHDCPARM. The updated source is found in the RHDCPARM member of the PPOPTION library. Because RHDCPARM was assembled and linked during the installation, SMP/E knows how to assemble and link it. All load modules containing RHDCPARM are automatically relinked.

```
++USERMOD(UMODPRM)         /* UNIQUE USERMOD NAME            */.
++VER(Z038) FMID(CGJF000)  /* RHDCPARM BELONGS TO CA-IDMS/DB */.
++SRC(RHDCPARM) TXLIB(PPOPTION) /* SOURCE IS IN PPOPTION     */.
```

If you want to assemble a source module that is not in the PPOPTION library, check the DISTSRC library and copy it to PPOPTION.

The example did not create a new (additional) source module or an additional load module; existing ones were replaced. Typically, USERMODs that create new source modules also create new load modules, and vice versa; these USERMODs are more complex. Creating a new source module whose name is unknown to SMP/E requires JCLIN statements (see 13.6, “JCLIN” on page 13-9) and a distribution library, even though the USERMOD is never ACCEPTed.

```
++SRC(FETABCV7) DISTLIB(DISTSRC) TXLIB(PPOPTION).
```

13.5 Element Updates

++ZAP, ++MACUPD and ++SRCUPD statements are used to modify elements. Typically, USERMODs are not required to update an element.

Example:

```
++ZAP(IDMSARBK).
  NAME IDMSARBK
  VER  0007B8 9002,0C22
  REP  0007B8 9002,8C22
```

13.6 JCLIN

Most USERMODs do not need JCLIN statements; a USERMOD to change a macro parm value does not require JCLIN. ++JCLIN statements are required for:

- Creating a new load module
- Changing the link edit characteristics (link edit control cards) of an existing load module

The ++JCLIN statement must follow the ++VER statement and precede any element statements. JCLIN is usually inline after the ++JCLIN statement.

Example:

```
++JCLIN.
//LNKNRMT  EXEC PGM=IEWL,PARM='LET,MAP,LIST,NCAL'
//DISTLOAD DD DSN=DO.NOT.CHANGE.DISTLOAD,DISP=SHR
//SYSLMOD  DD DSN=DO.NOT.CHANGE.LOADLIB,DISP=SHR
//SYSLIN   DD *
 ORDER   DFHEAI
 INCLUDE DISTLOAD(DFHEAI)
 INCLUDE DISTLOAD(CV1INTC)
 INCLUDE DISTLOAD(IDMS)
 INCLUDE DISTLOAD(DFHEAI0)
 INCLUDE DISTLOAD(IDMSTRUE)
 ENTRY   STARTUP
 MODE    AMODE(31),RMODE(24)
 NAME    CV1INTC(R)
```

JCLIN is confusing, because it *appears* to be an executable link. However, SMP/E does not execute these. It uses them to update the CSI. This is the JCLIN processing referred to in the output listing of APPLY jobs. After the JCLIN processing is successfully completed, SMP/E processes the element statements (++SRC, ++MOD, etc.). SMP/E uses the element statements to determine how to generate the link edit control cards that are passed to the linkage editor. So, if a USERMOD does not contain any element statements, the CSI is updated with the new link changes (JCLIN processing) but no link edit cards are passed to the linkage editor and no link is performed.

Below are the rules for coding JCLIN. The result looks invalid because; named datasets do not exist, modules are not in the libraries from which they are being included, DD statements such as SYSPRINT are missing, etc. None of this matters because these are not the images that SMP/E passes to the linkage editor.

- **EXEC Statement**. The PARM operand must specify whether the load module is reentrant, refreshable, or reusable (RENT, REFR, or REUS). Other parameters are ignored because their values are taken from options established in the global zone.
- **INCLUDE Statements**. The INCLUDEd module's distribution load library:
  - DISTLOAD for CA-IDMS
  - CWIF0LLD for CA-IDMS Tools

- CABF0LLD for CA-Endevor/DB

Each library specified on an INCLUDE statement must match the ddname and low-level qualifier of a DD statement in the JCLIN. The ddname must match a DDDEF defined during the install.

- **Distribution Load Library DD Statement**. The low-level qualifier for the dataset name matches the ddname.
- **SYSLMOD DD Statement**. The low-level qualifier for the SYSLMOD statement should be the ddname of a target load library (defined in the DDDEFs). Valid low-level qualifiers include:
  - LOADLIB — CA-IDMS
  - CWIF0LLT — CA-IDMS Tools
  - CABF0LLT — CA-Endevor/DB
  - APFLIB — CA-IDMS
  - INTBLOAD — CA-IDMS

    Note: The dsname of your loadlib is unimportant and does not have to match the dataset name on the SYSLMOD DD statement.

Example: You want to create a new load module in your target loadlib IDMS.LOAD.

```
//SYSLMOD DD DSN=GARBAGE.GARBAGE.LOADLIB,DISP=SHR
```

Explanation: This is correct because the DDDEF for the target loadlib is LOADLIB.

```
//SYSLMOD DD DSN=IDMS.LOAD,DISP=SHR
```

Explanation: This is incorrect and results in an APPLY error because there is no DDDEF called LOAD.

Note: You can create more DDDEFs or add DD statements to the APPLY JCL if you want the load module created in a different loadlib.

13.7 JCL to Install USERMODs

USERMODs should be RECEIVEd, APPLYed CHECK, and APPLYed. These commands can be executed in three separate steps; add condition code checking on the last two steps so they are executed only when the prior steps are successful.

```
//RECEIVE  EXEC PGM=GIMSMP,PARM='DATE=U'
⋮
//APPCHECK EXEC PGM=GIMSMP,PARM='DATE=U',COND=(0,LT)
⋮
//APPLY    EXEC PGM=GIMSMP,PARM='DATE=U',COND=(0,LT)
```

Each step requires:

- SMPCSI DD statement for the CSI
- SMPCNTL statement for the SMP/E commands (RECEIVE, APPLY)

If you are using an SMP/E proc, it probably has a SYSIN DD statement instead of an SMPCNTL DD statement; you do not need to add an SMPCSI DD statement. Other statements would be required or not as discussed below.

CAUTION:
Never ACCEPT your USERMODs.

13.7.1 RECEIVE Step

The RECEIVE step requires a SMPPTFIN DD statement for the location of the USERMOD (the modification control statements). The data for SMPPTFIN DD statement can be located in:

- A sequential dataset
- A PDS member
- Inline data

The SMPPTFIN DD statement may point to one or more concatenated USERMODs. RECEIVE processing finds the selected USERMOD, scans it for syntax errors, and, if successful, copies it into a member in the SMPPTS dataset.

13.7.2 APPLY Steps

APPLY processing uses the copy of the USERMOD in the SMPPTS dataset; an APPLY step does not require a SMPPTFIN DD statement. The APPLY steps must access the libraries containing the input and output modules. The input modules are the elements described on the ++SRC statements, ++MOD statements, etc. The output modules are the load modules that are created, replaced or updated.

13.7.2.1 Input Libraries

The TXLIB parameter in the USERMOD identifies the input library; this is usually PPOPTION. Since the CSI contains a DDDEF for PPOPTION, no additional DD statements are required.

No additional DD statements are required in the APPLY steps for this USERMOD:

```
++SRC(#SVCOPT) TXLIB(PPOPTION).
```

13.7.2.2 Output Libraries

The output libraries are the loadlibs containing the updated or created load modules. The output libraries are identified by the low-level qualifier on the SYSLMOD DD statement. These are the typical load libraries and their associated product lines:

- LOADLIB — CA-IDMS
- CWIF0LLT — CA-IDMS Tools
- CABF0LLT — CA-Endevor/DB
- APFLIB — CA-IDMS
- INTBLOAD — CA-IDMS

These loadlibs have DDDEFs in the CSI and do not require additional DD statements.

If the output module is placed in another load library:

1. Refer to 13.6, “JCLIN” on page 13-9
2. Choose a unique ddname.
3. Use the ddname as a low-level qualifier on the SYSLMOD statement.
4. In the APPLY steps, add the DD Statement using the ddname and specify the dataset name.

Example: You want to create a load module in the DBA load library (DBA.LOADLIB) and the ddname is DBALOAD.

The SYSLMOD statement in the JCLIN specifies DBALOAD as the low-level qualfier.

```
//SYSLMOD DD DISP=SHR,DSN=DBALOAD
```

The APPLY steps contain a DD statement with a ddname of DBALOAD.

```
//DBALOAD DD DISP=SHR,DSN=DBA.LOADLIB
```

13.8 Installing USERMODs

If your USERMOD contains JCLIN, make a system backup of your SMP/E environment before installing it. Install the USERMOD using RECEIVE, APPLY CHECK and APPLY. If there are errors, remove the USERMOD by running RESTORE CHECK and RESTORE.

CAUTION:

- **Do not use REDO with USERMODs containing JCLIN.**
- **Do not ACCEPT your USERMODs.**
- **Do not run a RESTORE without running RESTORE CHECK first.**

It is possible to successfully install a USERMOD and verify it works properly, but encounter a hidden problem when you attempt to RESTORE the USERMOD. To avoid this situation, after installing a new USERMOD, immediately remove (RESTORE) it, and install the USERMOD again while systems backups are available, if needed.

13.8.1 MODID

If you get a MODID error during an APPLY CHECK of a USERMOD, the resolution depends on the type of SYSMOD causing the error.

- If the error occurs because the USERMOD does not have a prerequisite (PRE) or supersede (SUP) for a CA-IDMS maintenance PTF SYSMOD like GJ00SP3:

  1. REJECT the USERMOD
  2. Add a PRE for that PTF to the ++VER statement
  3. Reinstall it

  Warning: Do not add a SUP parameter.

- If the error occurs because an earlier site-specific USERMOD has a common element with the USERMOD you are installing, there are two solutions:

  – Remove both the USERMODs and combine them into one USERMOD.

  – REJECT the latest USERMOD and rename the element causing the problem. This requires renaming a member in the PPOPTION library. If the USERMOD contains JCLIN statements, you must change the INCLUDE statement.

13.9 REDO

There is a REDO parameter available on the APPLY command and you should not use it unless you understand the circumstances where it can cause problems.

- It is safe to use REDO on a SYSMOD that contains no JCLIN.
- It is not safe to use REDO on a SYSMOD containing JCLIN.

When you install a SYSMOD containing JCLIN, SMP/E:

- Automatically saves the original link information located in the CSI to the SMPSCDS dataset
- Changes the link information in the CSI

When you RESTORE a SYSMOD, SMP/E:

- Restores the original link information to the CSI using the SMPSCDS dataset

A RESTORE (similar to a ROLLBACK) returns all the information to its original state; it looks as if the SYSMOD was never applied.

Why does the REDO command cause problems for USERMODs created using JCLIN? The REDO parameter does not change SMP/E's APPLY processing:

- SMP/E copies the CSI's link information (from the first APPLY) to the SMPSCDS dataset (overlays the original link information)
- The CSI's link information is updated from the JCLIN statements.

The second APPLY does not fail, but SMP/E may not have enough information to successfully RESTORE the USERMOD; therefore, you may be unable to remove the USERMOD.

Instead of using the REDO option, if the USERMOD contains JCLIN:

1. Remove the USERMOD
2. Change the USERMOD
3. Reinstall the USERMOD

13.10 Restoring USERMODs

Remove USERMODs by running RESTORE CHECK and RESTORE. All of your USERMODs containing JCLIN must be removed before installing a maintenance tape, or other products in the same CSI, such as CA-IDMS Tools or CA-Endevor/DB, regardless if they are flagged by APPLY CHECK or not.

13.11 SUP Parameter

Never use the SUP parameter on the ++VER statement in your USERMODs.

13.12 MULTIPLE CVs

Multiple CVs can use one target loadlib if the customized modules for each CV are in separate DBA loadlibs. In the startup deck, the customized DBA loadlib is listed **before** the common target loadlib. The USERMODs to create the customized load modules must contain:

- A DD statement in the APPLY job to point to the DBA loadlib
- JCLIN where the SYSLMOD DD statement's low-level qualifier must match the ddname in the APPLY job

Each customized load module name must be unique because SMP/E cannot create different versions of the same load module, even if they are in different loadlibs. (Instead, SMP/E can maintain multiple copies of the same load module in different loadlibs.) If CA-IDMS requires the load module to have a specific name (for example, RHDCSRTT), you can specify an ALIAS for that name in the JCLIN. SMP/E creates the load module with the unique name, but CA-IDMS locates it via the alias name. An ALIAS statement in the JCLIN must have the entry point in parenthesis coded on the ALIAS statement. This is supported by the binder but not by the linkage editor. See the UMODSRTT member in SAMPJCL for an example.

13.13 GIM65901W

When installing a USERMOD containing JCLIN, you may get the following message:

```
GIM65901W MODULE aaaaaaaa WAS NOT INCLUDED IN LMOD
          bbbbbbbb BECAUSE ITS DISTRIBUTION ZONE cccc
          ddddddd IS DIFFERENT FROM THE TARGET ZONE
          cccc eeeeeee.
```

This typically means there is an APAR or PTF on module *aaaaaaaa* and the APAR or PTF has not been ACCEPTed.

Warning: SMP/E builds the new load module *bbbbbbbb* without including module *aaaaaaaa*. Although the condition code is only 4 (unless the link fails), SMP/E builds a deficient load module.

If *eeeeeee* is an APAR:

1. Remove the USERMOD
2. Run a LIST command (see D.1.19, “LIST Command” on page D-14) on the element to determine the APAR(s) causing the problem
3. Remove the APAR(s)
4. Reinstall the USERMOD
5. Reinstall the APARs

If *eeeeeee* is a maintenance tape PTF:

1. Remove the USERMOD
2. ACCEPT the maintenance
3. Reinstall the USERMOD

This type of error can be avoided by implementing a regular maintenance routine such as:

1. ACCEPT maintenance immediately
2. Backup the system after ACCEPTing the maintenance
3. Install all site-specific USERMODs
4. Install APARs, test fixes, optional APARs, etc.

Why is this advantageous?

1. There is a "clean" backup of the system without the maintenance applied. (This "backup" was created after APPLYing the previous maintenance tape.)
2. Refreshing the system from a backup is faster, cleaner, and easier than using SMP/E's RESTORE.
3. Immediately ACCEPTing maintenance prevents GIM65901W errors.

13.14 Multiple Load Module Versions (GIM41101E)

You may want to use USERMODs to create multiple new load modules which are variations on a theme.

Example: There are two DC/UCF systems, 120 and 150; and you want to create a new version of the RHDCCTAB load module for each system.

- The USERMODS are: CTAB120 and CTAB150.
- The load module names are: CTABV120 and CTABV150.
- Using the LIST command (D.1.19, "LIST Command" on page D-14), you determine module RHDCMT00 is linked into RHDCCTAB, therefore RHDCMT00 must be INCLUDEd in the new load modules. The MOD entry for RHDCMT00 before installing USERMODs:

```
RHDCMT00  LASTUPD      = CGJF000  TYPE=ADD
          LIBRARIES    = DISTLIB=DISTLOAD
          FMID         = CGJF000
          RMID         = CGJF000
          LMOD         = RHDCMT00
```

- After installing USERMOD CTAB120:

```
RHDCMT00  LASTUPD      = CTAB120  TYPE=UPD
          LIBRARIES    = DISTLIB=DISTLOAD
          FMID         = CGJF000
          RMID         = CGJF000
          LMOD         = RHDCMT00  CTABV120
```

- After installing USERMOD CTAB150:

```
RHDCMT00  LASTUPD      = CTAB150  TYPE=UPD
          LIBRARIES    = DISTLIB=DISTLOAD
          FMID         = CGJF000
          RMID         = CGJF000
          LMOD         = RHDCMT00  CTABV120  CTABV150
```

  The LMOD subentry is updated when RHDCMT00 is INCLUDEd in another load module (RHDCMT00 doesn't change). SMP/E updates the LASTUPD subentry automatically when a new load module containing RHDCMT00 is created.

  If you tried to remove CTAB120 you would get this message:

```
GIM41101E ** SMP/E DID NOT RESTORE MODULE ENTRY RHDCMT00 FOR SYSMOD
             CTAB120 BECAUSE MODULE RHDCMT00 WAS CHANGED BY CTAB150.
```

  SMP/E forces you to remove CTAB150 before modifying, removing, or reinstalling CTAB120. If these USERMODs affect different CVs, you must modify the loadlib used by one CV before modifying the loadlib for the CV you intended to affect.

This problem can be circumvented by coding USERMODS that separate JCLIN updates from other modifications. This requires:

1. Determine how many new load modules are related by a common INCLUDE statement. From the example above, there are two new load modules: CTABV120 and CTABV150.
2. Code a USERMOD containing JCLIN to create each new load module.
3. Install and remove each USERMOD separately to verify each one works correctly.
4. Create a new USERMOD containing the JCLIN from all of the USERMODs. Do not include their elements. For example:

```
++USERMOD(CTABJCL).
++VER(Z038) FMID(CGJF000).
++JCLIN.
//CV120    EXEC  PGM=IEWL,PARM='RENT'
//SYSLMOD  DD DSN=DBAL120,DISP=SHR
//DISTLOAD DD DSN=DO.NOT.CHANGE.DISTLOAD,DISP=SHR
//SYSLIN   DD   *
 ORDER   RHDCMT00
 INCLUDE DISTLOAD(RHDCMT00)
 INCLUDE DISTLOAD(CTABV120)
 MODE    AMODE(31),RMODE(ANY)
 ENTRY   MT00EP1
 ALIAS   RHDCMT00(MT00EP1)
 NAME    CTABV120(R)
//CV150    EXEC  PGM=IEWL,PARM='RENT'
//SYSLMOD  DD DSN=DBAL150,DISP=SHR
//DISTLOAD DD DSN=DO.NOT.CHANGE.DISTLOAD,DISP=SHR
//SYSLIN   DD   *
 ORDER   RHDCMT00
 INCLUDE DISTLOAD(RHDCMT00)
 INCLUDE DISTLOAD(CTABV150)
 MODE    AMODE(31),RMODE(ANY)
 ENTRY   MT00EP1
 ALIAS   RHDCMT00(MT00EP1)
 NAME    CTABV150(R)
/*
//
```

5. RECEIVE and APPLY the USERMOD containing the JCLIN. Because the USERMOD contains no elements (++SRC, ++MAC or ++MOD):
   - The RECEIVE job issues a warning, which can be ignored.
   - The CSI is updated but no assemblies or links are triggered.
6. Remove the JCLIN from the original USERMODs. This makes them elements-only USERMODs.
7. Install each elements-only USERMOD. For example:

```
++USERMOD(CTAB120).
++VER(Z038) FMID(CGJF000).
++SRC(CTABV120)
     DISTLIB ( DISTSRC  )
     DISTMOD ( DISTLOAD )
     TXLIB   ( PPOPTION ).
```

Because the second set of USERMODs contain no JCLIN, no LMOD subentries are added to the CSI, and you can install them with REDO to pick up changes.

Whenever you install one of these USERMODs, the ++SRC statements initiates the assemblies and links, even though the USERMOD contains no JCLIN, because SMP/E contains the link information (the element name in the ++SRC statement matches a MOD entry in the CSI containing an LMOD subentry).

In between maintenance cycles, do not RESTORE the elements-only USERMODs; this deletes the MOD entry in the CSI created by the JCLIN-only USERMOD.

Appendix A. CA-IDMS Product List

A.1 CA-IDMS Products A-4
A.1.1 CA-IDMS Base Products A-4
A.1.2 Tools Products in CA-IDMS Family A-4
A.1.3 CA-IDMS Transparency Products A-5
A.1.4 CA-IDMS TP Access Interfaces A-5
A.1.5 CA-IDMS/Culprit™ Interfaces A-5
A.2 CA-IDMS Tools Products A-7
A.3 CA-Endevor/DB A-8

This appendix describes the CA-IDMS product line.

The CA-IDMS product line consists of multiple integrated products which are installed together to create the CA-IDMS system environment. This Appendix lists the products which may be installed with the integrated installation procedure described in this manual.

The product mix contained on the installation tape may vary from one tape to the next. For the exact list of products contained on and installable from a particular genlevel tape, refer to the VARBLIST member in the SAMPJCL library. Product passwords are required for the installation of most products. The correct product passwords for your CA-IDMS product mix are included in your installation package.

A.1 CA-IDMS Products

A.1.1 CA-IDMS Base Products

Product Code	Product Name	Password Required	CA LMP Key Required
GJ	CA-IDMS/DB	Yes	Yes
FI	CA-IDMS/CICS Support	No	No
EJ	CA-IDMS/DBCS Option	No	Yes
GQ	CA-IDMS/DC	Yes	Yes
GS	CA-IDMS/DDS (Distributed Database System)	Yes	Yes
XS	CA-IDMS/Server	Yes	Yes
GT	CA-IDMS/SQL Option	Yes	Yes
GP	CA-IDMS/UCF	Yes	Yes
H8	Unicenter TNG CA-IDMS/Agent	Yes	Yes

A.1.2 Tools Products in CA-IDMS Family

Product Code	Product Name	Password Required	CA LMP Key Required
FE	CA-ADS	Yes	Yes
FE	CA-ADS ASF Option	Yes	No
FF	CA-ADS/BATCH	Yes	Yes
FK	CA-Culprit	Yes	Yes
FL	CA-EDP-Auditor™	Yes	Yes
FH	CA-ICMS	Yes	Yes
FM	CA-IDMS/APPC	Yes	Yes
A4	CA-IDMS/Dictionary Loader	Yes	Yes
GY	CA-IDMS/Performance Monitor	Yes	Yes
GX	CA-IDMS/Presspack	Yes	Yes

Product Code	Product Name	Password Required	CA LMP Key Required
FG	CA-OLQ	Yes	Yes
ZF	CA-VTX/PRESTEL	Yes	Yes
ZG	CA-VTX/TELETEL	Yes	Yes

A.1.3 CA-IDMS Transparency Products

Product Code	Product Name	Password Required	CA LMP Key Required
ZE	CA-IDMS/DBOMP Transparency	Yes	Yes
GU	CA-IDMS/DLI Transparency	Yes	Yes
GV	CA-IDMS/Total Transparency	Yes	Yes
GW	CA-IDMS/VSAM Transparency	Yes	Yes

A.1.4 CA-IDMS TP Access Interfaces

Product Code	TP Interfaces	Password Required	CA LMP Key Required
GP	CA-IDMS TP Option for CICS	Yes	No
XU	CA-IDMS TP Option for INTERCOM	Yes	Yes
QL	CA-IDMS TP Option for SHADOW	Yes	Yes
XV	CA-IDMS TP Option for TASKMASTER	Yes	Yes

A.1.5 CA-IDMS/Culprit™ Interfaces

Product Code	Interfaces	Password Required	CA LMP Key Required
FK	IMS/DL1 Interface	Yes	No
FK	INT-FCS	Yes	No

Product Code	Interfaces	Password Required	CA LMP Key Required
FK	INT-LIBRARIAN	Yes	No
FK	INT-LIFE70	Yes	No
FK	INT-PANVALET	Yes	No
FK	INT-RDMS	Yes	No
FK	INT-TOTAL	Yes	No

A.2 CA-IDMS Tools Products

Product Code	Product Name	Password Required	CA LMP Key Required
O5	CA-IDMS/ADS Alive	Yes	Yes
ME	CA-IDMS/ADS Trace	Yes	Yes
H4	CA-IDMS/DB Analyzer	Yes	Yes
H7	CA-IDMS/DB Audit	Yes	Yes
IR	CA-IDMS/DB Reorg	Yes	Yes
O2	CA-IDMS/DC SORT	Yes	Yes
KJ	CA-IDMS/DML Online	Yes	Yes
OT	CA-IDMS/Database Extractor	Yes	Yes
MV	CA-IDMS/Dictionary Migrator	Yes	Yes
M3	CA-IDMS/Dictionary Module Editor	Yes	Yes
M4	CA-IDMS/Dictionary Query Facility	Yes	Yes
O3	CA-IDMS/Enforcer	Yes	Yes
FP	CA-IDMS/Journal Analyzer	Yes	Yes
F7	CA-IDMS/Log Analyzer	Yes	Yes
F8	CA-IDMS/Master Key	Yes	Yes
F9	CA-IDMS/Online Log Display	Yes	Yes
O4	CA-IDMS/SASO	Yes	Yes
GI	CA-IDMS/Schema Mapper	Yes	Yes
O6	CA-IDMS/Task Analyzer	Yes	Yes

A.3 CA-Endevor/DB

Product Code	Product Name	Password Required	CA LMP Key Required
HE	CA-Endevor/DB	Yes	Yes

Appendix B. Installation Tape Description

B.1 Installation Tape Format B-3
B.2 Installation Tape Contents B-4

B.1 Installation Tape Format

This appendix describes the installation tape.

The machine readable program materials required for installation are distributed on a multifile tape. The tape is available in the following formats:

- 6250 bpi reel
- 3480 cartridge
- 3490 cartridge

The format of the tape requires a combination of standard OS/390 utilities and Computer Associates utilities to perform the installation.

The volume serial number is **F0GJsB**, where:

- **F0** represents Release 15.0
- **GJ** is the Computer Associates two-character product ID for CA-IDMS
- **s** is the service pack
- **B** indicates a base tape

B.2 Installation Tape Contents

A list and description of each file contained on the installation tape is provided below.

File	Dataset name	Format	Description
1	CAI.INSTALL	IEBCOPY	Unused file
2 through 8	CAI.**F0GJsB**.FILE02 CAI.**F0GJsB**.FILE08		Unused files
9	CAI.IDMS.SAMPJCL	IEBCOPY	CA-IDMS Sample customization JCL
10	CAI.IDTOOLS.SAMPJCL	IEBCOPY	CA-IDMS Tools Sample customization JCL
11	CAI.CAEDB.SAMPJCL	IEBCOPY	CA-Endevor/DB Sample customization JCL
12	CAI.**F0GJsB**.FILE12	IEBGENER	Service Pack Identifier
13	CAI.**F0GJsB**.FILE13	QUTBTCUT	CAIIPDS boot file
14	CAI.**F0GJsB**.FILE14	QUTBTCUT	Encrypted object modules for CA-IDMS
15	CAI.**F0GJsB**.FILE15	QUTBTCUT	Encrypted source modules for CA-IDMS
16	CAI.**F0GJsB**.FILE16	QUTBTCUT	Encrypted object modules for CA-IDMS Tools
17	CAI.**F0GJsB**.FILE17	QUTBTCUT	Encrypted source modules for CA-IDMS Tools
18	CAI.**F0GJsB**.FILE18	QUTBTCUT	Encrypted object modules for CA-Endevor/DB
19	CAI.**F0GJsB**.FILE19	QUTBTCUT	Encrypted source modules for CA-Endevor/DB
20	CAI.**F0GJsB**.FILE20	IEBGENER	CAISAG program for CA-IDMS
21	CAI.**F0GJsB**.FILE21	IEBGENER	CAISAG skeleton for CA-IDMS
22	CAI.**F0GJsB**.FILE22	IEBGENER	CAISAG program for CA-IDMS Tools
23	CAI.**F0GJsB**.FILE23	IEBGENER	CAISAG skeleton for CA-IDMS Tools
24	CAI.**F0GJsB**.FILE24	IEBGENER	CAISAG program for CA-Endevor/DB
25	CAI.**F0GJsB**.FILE25	IEBGENER	CAISAG skeleton for CA-Endevor/DB
26	CAI.**F0GJsB**.FILE26	IEBCOPY	Migration utility object file
27	CAI.**F0GJsB**.FILE27	IEBGENER	SPG text for CA-IDMS/SASO
28	CAI.**F0GJsB**.FILE28	IEBGENER	IDMSDIRL input data file
29	CAI.**F0GJsB**.FILE29	IEBGENER	CA-Culprit test file
30	CAI.**F0GJsB**.FILE30	IEBCOPY	USMFLCON load library

File	Dataset name	Format	Description
31	CAI.**F0GJsB**.FILE31	IEBCOPY	USMFLCON source library
32	SMPMCS	IEBGENER	SMP/E control file for CA-IDMS
33 and up		IEBCOPY	SMP/E relfiles and control files for CA-IDMS/Tools and CA-Endevor/DB

Appendix C. VARBLIST

C.1 CA-IDMS VARBLIST Member Listing C-3
C.2 CA-IDMS Tools VARBLIST Member Listing C-23
C.3 CA-Endevor/DB VARBLIST Member Listing C-39

C.1 CA-IDMS VARBLIST Member Listing

This appendix shows the first VARBLIST member delivered for CA-IDMS Release 15.0. The VARBLIST member is likely to change at each genlevel.

CAUTION:
Running the CAISAG program from one genlevel with an already-customized VARBLIST from a prior genlevel causes unpredictable results.

```
GENLEVEL = F00103DOA00 Do not modify this line
*************************************************************************
***                                                                   ***
***    This is the list of variables that will be used for input      ***
***    by the SPECIFY AND GENERATE program.  Customize this           ***
***    member (VARBLIST), the PASSWORD member, the JOBCARD            ***
***    member and the SETUP member (if needed) before submitting      ***
***    the CAISAG job.                                                ***
***                                                                   ***
***    The values in this list will be translated to uppercase        ***
***    unless they are enclosed in single quotes.                     ***
***                                                                   ***
***    There are three valid types of lines in this list:             ***
***          a comment (asterisk in column 1)                         ***
***          a completely blank line                                  ***
***          a variable line                                          ***
***                                                                   ***
***    Variable lines consist of three fields:  a variable            ***
***    name, an equals sign, and a value.  If there are any           ***
***    blanks in the value you specify, enclose the value in          ***
***    single quotes.  Any characters following the value             ***
***    are ignored.                                                   ***
***                                                                   ***
***    For a quick check of your updates, enter the following         ***
***    three commands:                                                ***
***             EX ALL               Exclude all                     ***
***             F ALL P'<' 1 25      Find lowercase columns 1-25     ***
***             EX ALL '*' 1         Exclude all comments            ***
***                                                                   ***
***    In the PASSWORD member, overtype the default passwords         ***
***    with the passwords supplied on your Product Authorization      ***
***    sheets.  Any unused password lines can be deleted.             ***
***                                                                   ***
*************************************************************************
```

```
*-------------------------------------------------------------------*
*                                                                   *
*      Following are the products that can be installed from        *
*      this tape.  Specify either 'NO' or 'INSTALL' for each        *
*      product.                                                     *
*                                                                   *
*      If you are doing a base install (as opposed to an ADDON      *
*      install), you must specify 'INSTALL' for CA-IDMS/DB.         *
*      CA-IDMS/DB is a required core product and includes DB,       *
*      IDD and CV.                                                  *
*                                                                   *
*      If you install CA-IDMS/DC, also install CA-IDMS/UCF.         *
*      You may install CA-IDMS/UCF without installing CA-IDMS/DC.   *
*                                                                   *
*      If you have already done a base install and now you          *
*      are installing an additional product, that is an ADDON       *
*      install.  For an ADDON install, specify 'INSTALL' for        *
*      the additional product and 'NO' for all the other products.  *
*                                                                   *
*-------------------------------------------------------------------*
CA-IDMS/DB                      = INSTALL
CA-IDMS/DC                      = NO
CA-IDMS/UCF                     = NO
TSO INTERFACE FOR CA-IDMS/UCF   = NO
TP/CICS                         = NO      REQUIRES CA-IDMS/CICS
CA-IDMS/DB-SQL                  = NO      SQL OPTION
CA-IDMS/PERFORMANCE MONITOR     = NO
CA-IDMS DICTIONARY LOADER       = NO
CA-PRESSPACK                    = NO
CA-ADS                          = NO
CA-ADS/BATCH                    = NO      REQUIRES CA-ADS
CA-OLQ                          = NO
CA-ADS/APPC                     = NO      REQUIRES CA-ADS
ASF-OPTION                      = NO      REQUIRES CA-ADS
CA-EDP/AUDITOR                  = NO      REQUIRES CA-IDMS/CULPRIT
CA-IDMS/DL1 TRANSPARENCY        = NO
CA-IDMS/TOTAL TRANSPARENCY      = NO
CA-IDMS/VSAM TRANSPARENCY       = NO
CA-IDMS/DBOMP TRANSPARENCY      = NO
CA-IDMS TP OPTION FOR SHADOW    = NO
CA-IDMS TP OPTION FOR TASKMASTER = NO
CA-ICMS                         = NO      REQUIRES CA-ADS AND ASF-OPTION
CA-IDMS/CULPRIT                 = NO
CA-IDMS CICS SUPPORT            = NO
CA-IDMS/DDS                     = NO
CA-IDMS/DBCS                    = NO
CA-VTX/PRESTEL                  = NO
CA-VTX/TELETEL                  = NO
CA-IDMS/SERVER                  = NO
CA-IDMS/CMS OPTION              = NO
CA-IDMS TP OPTION FOR INTERCOMM  = NO
CA-UNICENTER TNG CA-IDMS AGENT   = NO
```

```
*--------------------------------------------------------------------*
*                                                                    *
*    The jobs which are generated will be output as separate         *
*    members.  The last two characters of the member names will      *
*    be ascending numbers.  You can specify the beginning of the     *
*    member names, up to 6 characters.  If you specify JOB, for      *
*    example, the member names will be JOB01, JOB02, JOB03 etc.      *
*                                                                    *
*--------------------------------------------------------------------*
BEGINMEM = JOB

*--------------------------------------------------------------------*
*                                                                    *
*    There are two ways this tape can be installed.  You can install *
*    a completely new 15.0 system, keeping nothing from your 12.0,   *
*    14.0 or 14.1 system, or you can install only the 15.0 software, *
*    retaining your 12.0, 14.0 or 14.1 dictionaries.  The latter is  *
*    called an upgrade install.                                      *
*                                                                    *
*    If converting from a CA-IDMS release prior to 12.0, you may     *
*    NOT do an upgrade.                                              *
*                                                                    *
*    When doing an upgrade install, you cannot install additional    *
*    products.  Install only the products that were installed on     *
*    your earlier system.  Then do an ADDON install for any          *
*    additional product(s).                                          *
*                                                                    *
*--------------------------------------------------------------------*
GJUPGRAD = NO          Specify 'YES' to do an upgrade
*                      Specify 'NO' to do a complete install
```

```
*-----------------------------------------------------------------------*
*                                                                       *
*      The following values will be used as defaults for datasets       *
*      which will be allocated in this install.  These are only         *
*      defaults and can be changed in the variables which use them.     *
*                                                                       *
*      For example, any of the following are valid:                     *
*                                                                       *
*           PYCSIDSN = @DSPREFIX@.CSI        use value as shown         *
*           PYCSIDSN = @DSPREFIX@.vsam.csi   change last part           *
*           PYCSIDSN = vsam.idms.CSI         change first part          *
*           PYCSIDSN = vsam.IDMS.R150.csi    change whole value         *
*                                                                       *
*-----------------------------------------------------------------------*
DSPREFIX  = IDMS.R150   Dataset prefix for other files
DBPREFIX  = IDMS.TEST   Dataset prefix for database files
UNIT      = SYSDA       Unit used for permanent disk datasets
PACK      = IDMS01      Disk VOLSER used for permanent datasets
SBLK      = 8880        Blocksize used for source libraries
LBLK      = 18452       Blocksize used for load libraries


*-----------------------------------------------------------------------*
*
*    Code the variables below to conform to your site's conventions.  *
*    These values will be used in the generated JCL.                  *
*
*-----------------------------------------------------------------------*
GJSYSOUT = *            Specify your sysout class, for example: A
GJWRKVOL = ''           DASD VOLSER for temporary work files, optional
GJWRKUNI = SYSDA        DASD unit for temporary work files
GJTAPUNI = TAPE         Tape unit
TAPCLASS = T            Job class for tape jobs
GJCOBNAM = IGYCRCTL     COBOL compiler name (IKFCBL00 for VS COBOL)
GJLNKNAM = IEWL         Specify the name of your link edit program
GJASMHLA = YES          Are you using IBM's High Level Assembler
GJASMNAM = ASMA90       Specify the program name of your assembler
GJZAPNAM = AMASPZAP     Specify the program name of your zap utility
GJBTAM   = NO           Does your site use BTAM? (YES/NO) (requires CA-IDMS/DC)
GJTCAM   = NO           Does your site use TCAM? (YES/NO) (requires CA-IDMS /DC)
GJSAMDSN = @DSPREFIX@.SAMPJCL DSNAME of this SAMPJCL library
```

```
*--------------------------------------------------------------------*
*                                                                    *
*    CA-IDMS is installed with SMP/E.   SMP/E will maintain          *
*    information about CA-IDMS in a vsam file called the CSI         *
*    (Consolidated Software Inventory).  The files, modules and      *
*    utilities known to the CSI comprise the SMP/E environment.      *
*                                                                    *
*    We recommend that you install CA-IDMS in its own SMP/E          *
*    environment, i.e., in its own CSI, and specify YES below.       *
*    Specifying YES will cause a new CSI and SMP/E environment       *
*    to be allocated.                                                *
*                                                                    *
*--------------------------------------------------------------------*
GJNEWCSI = YES          Use NO only if you want to install CA-IDMS
into another product's existing CSI

GJSMPLEV = YES          Is your release of SMP/E V1R8.1 or higher?  (YES/NO)
*                       (It is shown in every SMP/E listing, for example,
*                       GIMSMP LVL 18.1 or SMP/E IS AT LEVEL 25.21.)

*    For a new SMP/E environment, the datasets below will be allocated.
*    For an existing environment, it will be assumed that the specified
*    datasets already exist.

PYCSIDSN = @DSPREFIX@.CSI                              Dsname of CSI
PYPTSDSN = @DSPREFIX@.SMPPTS                           Dsname of SMPPTS
GJLTSDSN = @DSPREFIX@.SMPLTS                           Dsname of SMPLTS
PYMTSDSN = @DSPREFIX@.SMPMTS                           Dsname of SMPMTS
PYSTSDSN = @DSPREFIX@.SMPSTS                           Dsname of SMPSTS
PYSDSDSN = @DSPREFIX@.SMPSCDS                          Dsname of SMPSCDS
GJSMPLOG = @DSPREFIX@.SMPLOG                           Dsname of SMPLOG
GJALTLOG = @DSPREFIX@.SMPLOGA                          Dsname of SMPLOGA


*--------------------------------------------------------------------*
*                                                                    *
*    The following variables are ignored if installing into an       *
*    existing SMP/E environment (GJNEWCSI=NO).                       *
*                                                                    *
*--------------------------------------------------------------------*
GJCSIUNI = @UNIT@       Disk unit for new CSI
GJCSISER = @PACK@       Disk VOLSER for new CSI
GJPTSUNI = @UNIT@       Disk unit for SMPPTS
GJPTSSER = @PACK@       Disk VOLSER for SMPPTS
GJPTSBLK = @SBLK@       Blocksize for SMPPTS
GJLTSUNI = @UNIT@       Disk unit for SMPLTS
GJLTSSER = @PACK@       Disk VOLSER for SMPLTS
GJLTSBLK = @LBLK@       Blocksize for SMPLTS
GJMTSUNI = @UNIT@       Disk unit for SMPMTS
GJMTSSER = @PACK@       Disk VOLSER for SMPMTS
GJMTSBLK = @SBLK@       Blocksize for SMPMTS
GJSTSUNI = @UNIT@       Disk unit for SMPSTS
GJSTSSER = @PACK@       Disk VOLSER for SMPSTS
GJSTSBLK = @SBLK@       Blocksize for SMPSTS
GJSDSUNI = @UNIT@       Disk unit for SMPSCDS
GJSDSSER = @PACK@       Disk VOLSER for SMPSCDS
GJSDSBLK = @SBLK@       Blocksize for SMPSCDS
GJLOGUNI = @UNIT@       Disk unit for SMP/E logs
```

```
GJLOGSER = @PACK@       Disk VOLSER for SMP/E logs
GJDSPREF = @DSPREFIX@                  High-level qualifiers (up to 28
*                                      characters) used for relfile
*                                      processing.


*-------------------------------------------------------------------*
*                                                                   *
*    The following variables are specific to the CA-IDMS system.    *
*    They are required for any combination of products you are      *
*    installing.                                                    *
*                                                                   *
*-------------------------------------------------------------------*
GJGLDMCL = R150DMCL     Name to be used for the global DMCL
*                       (If you are doing an upgrade, use the name
*                       of your current global DMCL.)
*
GJGLDBTB = R150DBTB     Name to be used for the DB name table
*
GJSVCNUM = 173          SVC number
GJCVNUM  = 150          Central version number for the DC system
GJDCSYS  = 150          CA-IDMS/DC system number
GJFRESTG = 1200         FREESTG value for assembly of #DCPARM
GJOPTCNT = YES          Parameter used in assembly of IDMSOPTI
*                            GJOPTCNT=YES means central or local
*                            GJOPTCNT=ONLY means exclusively central
*                            GJOPTCNT=NO means exclusively local

GJACMETH = BDAM         Disk file access method (BDAM or VSAM)
*                       BDAM is recommended

GJCASE   = MIXED        This variable is for European sites. Specify
*                       MIXED unless your system has terminals which
*                       cannot print mixed case characters.  (MIXED/UPPER)

GJXARENT = 2000         Specify the value for the XA reentrant pool.
*                       The upper limit is 16383.  Do not put a comma
*                       in the number.

GJAUTH   = ''           Authorized userid for dictionary signon, if required
GJAUTHPW = ''           Authorized userid's password, if required

GJWTOEX  = YES          Do you want to link a customized WTOEXIT
*                       into your startup module?
GJSTUP   = ''           Optionally, specify a new name for your startup
*                       module (default names are IDMSDC and IDMSDCU)

*    GJPDETYP determines whether CA-IDMS product modules which
*    are eligible to use null PDEs will be defined during sysgen
*    processing (maps, dialogs and tables).
*
*    GJPDETYP determines whether CA-IDMS will allocate program
*    definition elements (PDEs) at runtime (dynamic) or at
*    system startup (static).
*
```

```
*    If DYNAMIC is chosen, program definitions for eligible
*    modules will not be added to the sysgen.  This will reduce
*    system startup storage and will allow the Central Version to
*    use null PDEs for eligible modules.  It allows PDEs for
*    maps, dialogs and tables to be loaded in XA storage.
*
*
*    If STATIC is chosen, all modules will be added to the sysgen
*    as PDEs, using more storage at startup.
*
*    For more information, refer to the CA-IDMS Installation and
*    Maintenance Guide, Chapter 5.

GJPDETYP = DYNAMIC      (DYNAMIC/STATIC)


*    In the sysgen, programs can be defined for storage protection.
*    The STORPROT variable determines whether PROGRAM statements
*    in the sysgen will have PROTECT or NOPROTECT.
*
*    If STORPROT=YES, then all PROGRAM statements will have PROTECT
*    and the CV SYSTEM statement will have NOPROTECT.  This is the
*    sysgen compiler default, and is a good choice for test systems.
*    This choice installs DLODxxxx members into the SYSTEM dictionary.
*
*    If STORPROT=NO, then all PROGRAM statements will have NOPROTECT
*    and the CV SYSTEM statement will have PROTECT.  This is a good
*    choice for a production system.  This choice installs DNODxxxx
*    members into the SYSTEM dictionary.
*
*    DLODxxxx members contain sysgen source using PROTECT on the
*    PROGRAM statements.  DNODxxxx members contain sysgen source
*    using NOPROTECT on the PROGRAM statements.
*
*    No matter which value you specify for the STORPROT variable
*    below, both DLODxxxx members and DNODxxxx members will be
*    installed by SMP/E into your DISTSRC library.  The value you

STORPROT = YES          Use storage PROTECT for programs (DLOD members)


*-------------------------------------------------------------------*
*                                                                   *
*    The following three variables control a tape backup for the    *
*    SMP/E environment.                                             *
*                                                                   *
*    If 'NO' is specified for GJSMPBKP, no JCL for the backup of    *
*    the SMP/E environment will be generated.                       *
*                                                                   *
*    The GJSBTAPE variable can specify a VOLSER name or can be null.  *
*    If it is null, no VOLSER will be generated in the DD statement   *
*    that points to the backup tape.  The operator can mount a      *
*    scratch tape.                                                  *
*                                                                   *
*    If the GJINITT variable is 'YES', an IEHINITT step will be     *
*    generated to initialize the tape.                              *
*                                                                   *
*-------------------------------------------------------------------*
GJSMPBKP = YES          Generate JCL to do an SMP/E backup (YES/NO)
GJSBTAPE = IDMS15       VOLSER of the backup tape, or null ('')
GJINITT  = YES          Initialize tape to the specified VOLSER (YES/NO)
```

```
*--------------------------------------------------------------------*
*                                                                    *
*    The following variables control the three tape backup/restore   *
*    steps for the database files.  The first is for the SYSDIRL     *
*    segment.  The second is for System 99.  The third is a final    *
*    one for the installed database.                                 *
*                                                                    *
*--------------------------------------------------------------------*

*     If specific VOLSERs are specified for the first three variables,
*     the three tapes should be initialized before submitting the
*     installation JCL.  If you leave the VOLSERs null as shown, a
*     specific volume will not be requested in the JCL and the
*     operator can mount a scratch volume.  (The restore steps do
*     not specify any VOLSERs; they use referbacks to the backup
*     steps.)  If 'NO' is specified for the VOLSER, neither the
*     backup JCL nor the restore JCL is generated and the BACKUP/
*     RESTORE action is ignored.

GJBK1VOL = ''           Tape VOLSER for backup of SYSDIRL segment
GJBK1ACT = RESTORE      RESTORE = backup & restore; BACKUP = backup only
GJBK2VOL = ''           Tape VOLSER for backup of System 99
GJBK2ACT = RESTORE      RESTORE = backup & restore; BACKUP = backup only
GJBK3VOL = ''           Tape VOLSER for backup of installed database
GJBK3ACT = RESTORE      RESTORE = backup & restore; BACKUP = backup only
GJBK1DSN = IDMSTAPE.BAK1DIRL  Dataset name for backup of SYSDIRL segment
GJBK2DSN = IDMSTAPE.BAK2DICT  Dataset name for backup of System 99
GJBK3DSN = IDMSTAPE.BAK3DICT  Dataset name for backup of installed database

*     Most sites do not use BLP processing and should not change
*     the following variables.

GJBLP    = NO           YES only if you need to use BLP processing
GJBLPHLQ = CAI          High level qualifier of tape file DSN for BLP


*--------------------------------------------------------------------*
*                                                                    *
*   The following parameters are tape file parameters used for       *
*   roll back, roll forward, backup and restore of the demo          *
*   database.  The JCL for these steps is not generated if the       *
*   value for GJTJRSER is 'NO'.                                      *
*                                                                    *
*   GJTJRSER is the VOLSER of the tape to be used for these steps.   *
*   If it is a null value (GJTJRSER = '') then the operator can      *
*   mount a scratch tape.  If a VOLSER is specified, then the        *
*   operator needs to mount a tape which has been initialized to     *
*   that VOLSER.  If GJTJRSER=NO, then tape journalling is not done. *
*                                                                    *
*--------------------------------------------------------------------*
GJTJRSER = ''           VOLSER of the tape used for journalling
GJTJRDSN = DEMOJRNL     Dsname for the tape journal file (file 1)
GJTJRDUM = DEMODUMP     Dsname for the tape backup file (file 2)
GJTJRDD  = SYSJRNL      DDname for tape journal
GJTJRUNI = TAPE         Tape unit for tape journal
GJTJRBLK = 23476        Blocksize for tape journal


*--------------------------------------------------------------------*
*                                                                    *
*   The next variable is only for sites that are installing          *
*   CA-IDMS/DC.  If you want to install a VTAM line, enter           *
*   a name for the VTAM APPLID.  If you leave a null value for       *
*   the GJVTAMID variable, no VTAM line will be installed.           *
*                                                                    *
*--------------------------------------------------------------------*

GJVTAMID = ''           VTAM application ID (APPLID), for example, IDMSDC1
```

```
*-------------------------------------------------------------------*
*                                                                   *
*   The next variable is for the ASF option only and is ignored     *
*   if you are not installing the ASF option.                       *
*                                                                   *
*-------------------------------------------------------------------*

*   The GJASFNIT variable allows you to bypass the format of the
*   ASFDICT and does not generate any of the DDDL steps.

GJASFNIT = YES          YES:  allocate and format a new ASF dictionary
*                       NO:   use existing ASF dictionary;  install only
*                             the software

*-------------------------------------------------------------------*
*                                                                   *
*   The next nine variables are for CA-IDMS/CICS Support and are    *
*   ignored if you are not installing it.                           *
*                                                                   *
*-------------------------------------------------------------------*
GJNTID   = DBDC         Next task ID (for TP/CICS)
GJCWA    = 0            CWA displacement
*                       This is the offset within the CICS CWA of a
*                       fullword to hold the address of the IDMSINTC
*                       module.  Only your CICS systems programming
*                       staff can assign this displacement.  It is a
*                       number of bytes, expressed in decimal.

GJCICXA  = YES          XA Support for CA-IDMS/CICS  (YES/NO)
GJTPINTC = ''           Specify the TP system name to be used for IDMSINTC
*                       If left null, the CICS System ID is used internally.
*
GJTPFET  = CICS         Specify the TP system name to be used for the
*                       front end table.  This cannot be null.
*
*    The following two variables work together.  If using SYSCTL,
*    or if no SVC is being used, specify 'NO' for GJCICSVC.  If
*    not using SYSCTL, specify the SVC number for GJCICSVC.  If
*    no SYSCTL file is specified, the SVC number is required.
*    If you specify an SVC number, you are limited to one SVC;
*    if you use 'NO', you are not.  If you specify an SVC
*    number, the SYSCTL value is not used.
GJCICCTL =  SYSCTL      SYSCTL DDname for CICS
GJCICSVC =  NO          Specify 'NO' or the SVC number (see comment above)

*
*    The following three variables identify datasets that will be
*    used for input.  The install does not update these libraries.
GJMCSDSN = CICS.V410.SDFHMAC                Name of your CICS maclib
GJCICDSN = CICS.V410.SDFHLOAD               Name of your CICS loadlib

*-------------------------------------------------------------------*
*                                                                   *
*   The next variables are for CA-IDMS/Performance Monitor and      *
*   are ignored if you are not installing it.                       *
*                                                                   *
*-------------------------------------------------------------------*
GJPMCOMP = 'COMPUTER ASSOCIATES INTERNATIONAL, INC.'
*               GJPMCOMP is the company name to appear on Perfmon reports
GJPMAACT = YES          Activate application monitor
GJPMALOG = YES          Write application monitor stats to DDLDCLOG
GJPMIACT = YES          Activate interval monitor
GJPMILOG = YES          Write interval monitor stats to DDLDCLOG
GJPMSTAT = YES          Write DC stats to DDLDCLOG
```

```
*-------------------------------------------------------------------*
*                                                                   *
*   The next variables are for CA-IDMS/CULPRIT and are ignored      *
*   if you are not installing it.                                   *
*                                                                   *
*-------------------------------------------------------------------*
GJIDDSEC =  NO          NO:  IDD security is turned off
*                       YES: IDD security is turned on

FKF001OP =  N           Install the FCS interface?  (Y/N)
GJFCSVER =  DL02        What is the version of FCS? (DL01/DL02)

FKF002OP =  N           Install the IMS interface?  (Y/N)
FKF003OP =  N           Install the CA-Librarian interface?  (Y/N)
FKF004OP =  N           Install the LIFE70 interface?  (Y/N)
FKF005OP =  N           Install the CA-Panvalet interface?  (Y/N)
FKF006OP =  N           Install the RDMS interface?  (Y/N)
FKF007OP =  N           Install the Total interface?  (Y/N)

*   Optionally, specify Culprit 15.0 profile CSECT Options
*   (limit is 55 characters)
GJCPRF1  =   ''
GJCPRF2  =   ''
GJCPRF3  =   ''
GJCPRF4  =   ''
GJCPRF5  =   ''
GJCPRF6  =   ''
GJCPRF7  =   ''
GJCPRF8  =   ''
GJCPRF9  =   ''
*-------------------------------------------------------------------*
*                                                                   *
*   The following parameters are used to create the CONVR102        *
*   member in SAMPJCL.  The CONVR102 job can be used to             *
*   to convert a 10.2 system to 15.0.                               *
*                                                                   *
*-------------------------------------------------------------------*
GJCNVLOD = IDMS.DMCC.CONV.LIB   Dsname of the loadlib to be allocated
GJCNVUNI = @UNIT@               Disk unit for the allocation
GJCNVSER = @PACK@               Disk VOLSER for the allocation
GJ102OBJ = IDMS.R102.OBJLIB     Dsname of your 10.2 objlib
GJBLKLIM = YES                  Is your linkage editor limited to objlibs
*                               blocksizes of 3200 or less?
```

```
*-------------------------------------------------------------------*
*                                                                   *
*   Specify the names of system datasets at your site.              *
*                                                                   *
*-------------------------------------------------------------------*
GJMACDSN = SYS1.MACLIB                       Maclib for assemblies
GJCOBCLB = SYS1.COBOL.LINKLIB                COBOL Compiler Steplib
GJCOBDSN = SYS1.VSCOBLIB                     COBOL Subroutine Library

*         Specify the name of your Unicenter TNG Framework for
*         OS/390 (Framework) loadlib if it is not in the OS/390
*         LNKLST.  This loadlib should contain CAIRIM.
GJCAIDSN = ''

*         If you are installing CA-ADS, specify the name of the
*         library that contains the C runtime modules.  It is
*         probably the Framework loadlib.
GJCDSN   = CAI.CAILIB

*         Specify the name of two IBM Language Environment libraries.
*         If you have these libraries, specify their names, even if
*         you want to run with COBOL II run-time libraries.  They are
*         required to assemble CEEUOPT and link RHDCLEFE.  See LI50756
*         for more information.
GJS37DSN = CEE.SCEELKED  Subroutine library (contains CEEBINT, CEEINT, etc.
GJM37DSN = CEE.SCEEMAC   This maclib will be used in the SYSLIB
                         concatenation for assemblies.

*         If you use TCAM or BTAM, specify the name of your
*         teleprocessing library, for example, SYS1.TELCMLIB.
GJTPDSN  = ''


*         If you are installing TP/CICS or CA-IDMS/VSAM
*         Transparency, you will need the IKJTCB or IEFJSSVT
*         macro, respectively.  If the required macro is not
*         in SYS1.MACLIB, specify the name of the library containing
*         it.  This library will be used in the SYSLIB concatenation
*         for assemblies.
GJMKJDSN = SYS1.AMODGEN

*         Most sites will not require the use of any
*         optional steplibs.  Specify them only if required
*         for link editor program, assembler program, LE
*         Cobol runtime loadlib, or other programs which
*         do not usually require a steplib.
GJSTEPL1 = ''                      First additional user steplib
GJSTEPL2 = ''                      Second additional user steplib
```

```
*--------------------------------------------------------------------*
*                                                                    *
*      Below are the variables for the datasets that will            *
*      be allocated.  For libraries, the variables specify           *
*      the dataset name, the unit, the VOLSER and the blocksize.     *
*      For database files, the variables specify the dataset         *
*      name, the unit and VOLSER, the starting page, the             *
*      pagesize and the number of pages.                             *
*                                                                    *
*--------------------------------------------------------------------*

*         Modules from the tape are downloaded into the
*         indirect source and object libraries.  They are
*         used as input by SMP/E.  The indirect libraries
*         can be deleted after the installation has been
*         completed.

GJINSDSN = @DSPREFIX@.INDSRC
GJINSUNI = @UNIT@
GJINSSER = @PACK@
GJINSSPC = CYL
GJINSPRI = 70
GJINSSEC = 5
GJINSDIR = 75
GJINSBLK = @SBLK@

GJINODSN = @DSPREFIX@.INDOBJ
GJINOUNI = @UNIT@
GJINOSER = @PACK@
GJINOSPC = CYL
GJINOPRI = 60
GJINOSEC = 5
GJINODIR = 300
GJINOBLK = 3120

*         The data to initialize the SYSDIRL dictionary is
*         downloaded from the tape into this dataset in Job 2.
*         It will be loaded into the dictionary in Job 8.  This
*         dataset can be deleted after the installation has
*         been completed.

GJDIRDSN = @DSPREFIX@.DIRLDATA
GJDIRUNI = @UNIT@
GJDIRSER = @PACK@
GJDIRSPC = TRK
GJDIRPRI = 96
GJDIRSEC = 15
GJDIRBLK = 4096

*         The DBA loadlib is a loadlib that will contain
*         the DBNAMES table and the DMCL.  The DBA loadlib
*         can be authorized.

GJDBADSN = @DBPREFIX@.DBA.LOADLIB
GJDBAUNI = @UNIT@
GJDBASER = @PACK@
GJDBASPC = CYL
GJDBAPRI = 2
GJDBASEC = 1
GJDBADIR = 8
GJDBABLK = @LBLK@
```

```
*         The PPOPTION library will contain customized source.

GJOLTDSN = @DSPREFIX@.PPOPTION
GJOLTUNI = @UNIT@
GJOLTSER = @PACK@
GJOLTSPC = CYL
GJOLTPRI = 2
GJOLTSEC = 1
GJOLTDIR = 25
GJOLTBLK = @SBLK@

*         The following libraries are CA-IDMS libraries
*         that will be updated by SMP/E during installation
*         and maintenance.

GJINBDSN = @DSPREFIX@.INTBLOAD        The INTBLOAD loadlib will
GJINBUNI = @UNIT@                     contain IDML and INTB102.
GJINBSER = @PACK@
GJINBSPC = TRK
GJINBPRI = 2
GJINBSEC = 1
GJINBDIR = 2
GJINBBLK = @LBLK@

GJAPFDSN = @DSPREFIX@.APFLIB          The APFLIB loadlib will contain
GJAPFUNI = @UNIT@                     GJF0INIT, RHDCSSFM, and the SVC
GJAPFSER = @PACK@                     load module.
GJAPFSPC = TRK
GJAPFPRI = 4
GJAPFSEC = 1
GJAPFDIR = 4
GJAPFBLK = @LBLK@

GJLLTDSN = @DSPREFIX@.LOADLIB         This is the main target loadlib.
GJLLTUNI = @UNIT@                     This loadlib should not be
GJLLTSER = @PACK@                     authorized.  Authorizing this
GJLLTSPC = CYL                        library may cause a S0C4 in
GJLLTPRI = 60                         Job 8.
GJLLTSEC = 5
GJLLTDIR = 300
GJLLTBLK = @LBLK@

GJSLDDSN = @DSPREFIX@.DISTSRC
GJSLDUNI = @UNIT@
GJSLDSER = @PACK@
GJSLDSPC = CYL
GJSLDPRI = 80
GJSLDSEC = 4
GJSLDDIR = 75
GJSLDBLK = @SBLK@

GJMLDDSN = @DSPREFIX@.DISTMAC
GJMLDUNI = @UNIT@
GJMLDSER = @PACK@
GJMLDSPC = CYL
GJMLDPRI = 30
GJMLDSEC = 3
GJMLDDIR = 100
GJMLDBLK = @SBLK@

GJLLDDSN = @DSPREFIX@.DISTLOAD
GJLLDUNI = @UNIT@
GJLLDSER = @PACK@
GJLLDSPC = CYL
GJLLDPRI = 75
GJLLDSEC = 5
GJLLDDIR = 375
GJLLDBLK = 8906
```

```
*         The SYSTEM SEGMENT parameters define the files and
*         areas that comprise the SYSTEM segment.  The files
*         in the SYSTEM segment define the runtime system as
*         well as the physical database.

GJSDMDSN = @DBPREFIX@.SYSTEM.DDLDML
GJSDMDD  = DCDML
GJSDMUNI = @UNIT@
GJSDMSER = @PACK@
GJSDMLOW = 1001         Starting page of SYSTEM.DDLDML area
GJSDMPSZ = 4276         Page size for area(s) in 'DCDML'
GJSDMNPG = 1000         Number of pages (blocks) in 'DCDML'

GJSLODSN = @DBPREFIX@.SYSTEM.DDLDCLOD
GJSLODD  = DCLOD
GJSLOUNI = @UNIT@
GJSLOSER = @PACK@
GJSLOLOW = 3001         Starting page of SYSTEM.DDLDCLOD area
GJSLOPSZ = 4276         Page size for area(s) in 'DCLOD'
GJSLONPG = 100          Number of pages (blocks) in 'DCLOD'

GJSLGDSN = @DBPREFIX@.SYSTEM.DDLDCLOG
GJSLGDD  = DCLOG
GJSLGUNI = @UNIT@
GJSLGSER = @PACK@
GJSLGLOW = 30001        Starting page of SYSTEM.DDLDCLOG area
GJSLGPSZ = 4276         Page size for area(s) in 'DCLOG'
GJSLGNPG = 4000         Number of pages (blocks) in 'DCLOG'

GJSRNDSN = @DBPREFIX@.SYSTEM.DDLDCRUN
GJSRNDD  = DCRUN
GJSRNUNI = @UNIT@
GJSRNSER = @PACK@
GJSRNLOW = 40001        Starting page of SYSTEM.DDLDCRUN area
GJSRNPSZ = 2676         Page size for area(s) in 'DCRUN'
GJSRNNPG = 1000         Number of pages (blocks) in 'DCRUN'

GJSSCDSN = @DBPREFIX@.SYSTEM.DDLDCSCR
GJSSCDD  = DCSCR
GJSSCUNI = @UNIT@
GJSSCSER = @PACK@
GJSSCLOW = 50001        Starting page of SYSTEM.DDLDCSCR area
GJSSCPSZ = 2676         Page size for area(s) in 'DCSCR'
GJSSCNPG = 2000         Number of pages (blocks) in 'DCSCR'
```

```
*         The CATSYS SEGMENT parameters define the files and
*         area that comprise the CATSYS segment.  This segment
*         is required for a CA-IDMS/DB install.

GJCCTDSN = @DBPREFIX@.CATSYS.DCCAT
GJCCTDD  = DCCAT
GJCCTUNI = @UNIT@
GJCCTSER = @PACK@
GJCCTLOW = 1            Starting page of CATSYS.DCCAT area
GJCCTPSZ = 4276         Page size for area(s) in 'DCCAT'
GJCCTNPG = 600          Number of pages (blocks) in 'DCCAT'

GJCCLDSN = @DBPREFIX@.CATSYS.DCCATLOD
GJCCLDD  = DCCATL
GJCCLUNI = @UNIT@
GJCCLSER = @PACK@
GJCCLLOW = 751          Starting page of CATSYS.DCCATLOD area
GJCCLPSZ = 4276         Page size for area(s) in 'DCCATL'
GJCCLNPG = 200          Number of pages (blocks) in 'DCCATL'

GJCCXDSN = @DBPREFIX@.CATSYS.DCCATX
GJCCXDD  = DCCATX
GJCCXUNI = @UNIT@
GJCCXSER = @PACK@
GJCCXLOW = 601          Starting page of CATSYS.DCCATX area
GJCCXPSZ = 4276         Page size for area(s) in'DCCATX'
GJCCXNPG = 100          Number of pages (blocks) in 'DCCATX'

*         The SYSDIRL SEGMENT parameters define the SYSDIRL.DDLDML
*         area.  The SYSDIRL segment contains the 15.0 IDMSNTWK
*         schema as well as all CULPRIT report source.  This segment
*         is required for a CA-IDMS/DB install.

GJDDMDSN = @DBPREFIX@.SYSDIRL.DDLDML
GJDDMDD  = DIRLDB
GJDDMUNI = @UNIT@
GJDDMSER = @PACK@
GJDDMLOW = 5001         Starting page of SYSDIRL.DDLDML area
GJDDMPSZ = 4276         Page size for area(s) in 'DIRLDB'
GJDDMNPG = 4000         Number of pages (blocks) in 'DIRLDB'

GJDLODSN = @DBPREFIX@.SYSDIRL.DDLDCLOD
GJDLODD  = DIRLLOD
GJDLOUNI = @UNIT@
GJDLOSER = @PACK@
GJDLOLOW = 4001         Starting page of SYSDIRL.DDLDCLOD area
GJDLOPSZ = 4276         Page size for area(s) in 'DIRLLOD'
GJDLONPG = 10           Number of pages (blocks) in 'DIRLLOD'

*         The SYSMSG SEGMENT parameters define the SYSMSG.DDLDCMSG
*         area.  The SYSMSG segment will contain all the Release
*         15.0 messages.  This segment is required for a CA-IDMS/DB
*         install.

GJMSGDSN = @DBPREFIX@.SYSMSG.DDLDCMSG
GJMSGDD  = DCMSG
GJMSGUNI = @UNIT@
GJMSGSER = @PACK@
GJMSGLOW = 10001        Starting page of SYSMSG.DDLDCMSG area
GJMSGPSZ = 4276         Page size for area(s) in 'DCMSG'
GJMSGNPG = 4000         Number of pages (blocks) in 'DCMSG'
```

```
*         The SYSLOC SEGMENT parameters define the SYSLOC.DDLOCSCR
*         area.  The SYSLOC segment is a local scratch area.  This
*         file is a temporary dataset for local batch processing.
*         it is defined to your DMCL but not allocated during the
*         install.

GJLSCDD  = DCLSCR
GJLSCLOW = 55001
GJLSCPSZ = 2676
GJLSCNPG = 2000

*         The APPLDICT SEGMENT parameters define the files that
*         comprise the APPLDICT segment.  This segment is the
*         application dictionary where schema, subschema and
*         application definitions are stored.

GJADMDSN = @DBPREFIX@.APPLDICT.DDLDML
GJADMDD  = DICTDB
GJADMUNI = @UNIT@
GJADMSER = @PACK@
GJADMLOW = 60001        Starting page of APPLDICT.DDLDML area
GJADMPSZ = 4276         Page size for area(s) in 'DICTDB'
GJADMNPG = 2000         Number of pages (blocks) in 'DICTDB'

GJALODSN = @DBPREFIX@.APPLDICT.DDLDCLOD
GJALODD  = DLODDB
GJALOUNI = @UNIT@
GJALOSER = @PACK@
GJALOLOW = 70001        Starting page of APPLDICT.DDLDCLOD area
GJALOPSZ = 4276         Page size for area(s) in 'DLODDB'
GJALONPG = 500          Number of pages (blocks) in 'DLODDB'

*         The SYSUSER SEGMENT parameters define the SYSUSER.DDLSEC
*         area.  The SYSUSER segment can contain any site-specific
*         user definitions. This segment is required for a CA-IDMS
*         install.

GJUSEDSN = @DBPREFIX@.SYSUSER.DDLSEC
GJUSEDD  = SECDD
GJUSEUNI = @UNIT@
GJUSESER = @PACK@
GJUSELOW = 48001        Starting page of SYSUSER.DDLSEC area
GJUSEPSZ = 4276         Page size for area(s) in 'SECDD'
GJUSENPG = 500          Number of pages (blocks) in 'SECDD'
```

```
*         The SYSSQL SEGMENT parameters define the files and
*         areas that comprise the SYSSQL segment.  The files in
*         the SYSSQL segment are required for sites installing
*         the SQL option, CA-IDMS/DB-SQL.

GJ$CTDSN = @DBPREFIX@.SYSSQL.DDLCAT
GJ$CTDD  = SQLDD
GJ$CTUNI = @UNIT@
GJ$CTSER = @PACK@
GJ$CTLOW = 20001        Starting page of SYSSQL.DDLCAT area
GJ$CTPSZ = 4276         Page size for area(s) in 'SQLDD'
GJ$CTNPG = 2000         Number of pages (blocks) in 'SQLDD'

GJ$CLDSN = @DBPREFIX@.SYSSQL.DDLCATL
GJ$CLDD  = SQLLOD
GJ$CLUNI = @UNIT@
GJ$CLSER = @PACK@
GJ$CLLOW = 25001        Starting page of SYSSQL.DDLCATL area
GJ$CLPSZ = 4276         Page size for area(s) in 'SQLLOD'
GJ$CLNPG = 500          Number of pages (blocks) in 'SQLLOD'

GJ$CXDSN = @DBPREFIX@.SYSSQL.DDLCATX
GJ$CXDD  = SQLXDD
GJ$CXUNI = @UNIT@
GJ$CXSER = @PACK@
GJ$CXLOW = 28001        Starting page of SYSSQL.DDLCATX area
GJ$CXPSZ = 4276         Page size for area(s) in 'SQLXDD'
GJ$CXNPG = 500          Number of pages (blocks) in 'SQLXDD'

*         The ASFDICT SEGMENT parameters define the files and
*         areas that comprise the ASFDICT segment.  This segment
*         is required only for sites installing the ASF option.

GJFDMDSN = @DBPREFIX@.ASFDICT.DDLDML
GJFDMDD  = ASFDML
GJFDMUNI = @UNIT@
GJFDMSER = @PACK@
GJFDMLOW = 80001        Starting page of ASFDICT.DDLDML area
GJFDMPSZ = 4276         Page size for area(s) in 'ASFDML'
GJFDMNPG = 2000         Number of pages (blocks) in 'ASFDML'

GJFDEDSN = @DBPREFIX@.ASFDICT.ASFDEFN
GJFDEDD  = ASFDEFN
GJFDEUNI = @UNIT@
GJFDESER = @PACK@
GJFDELOW = 83001        Starting page of ASFDICT.ASFDEFN area
GJFDEPSZ = 4276         Page size for area(s) in 'ASFDEFN'
GJFDENPG = 1000         Number of pages (blocks) in 'ASFDEFN'

GJFDADSN = @DBPREFIX@.ASFDICT.ASFDATA
GJFDADD  = ASFDATA
GJFDAUNI = @UNIT@
GJFDASER = @PACK@
GJFDALOW = 85001        Starting page of ASFDICT.ASFDATA area
GJFDAPSZ = 4276         Page size for area(s) in 'ASFDATA'
GJFDANPG = 2000         Number of pages (blocks) in 'ASFDATA'

GJFLODSN = @DBPREFIX@.ASFDICT.ASFLOD
GJFLODD  = ASFLOD
GJFLOUNI = @UNIT@
GJFLOSER = @PACK@
GJFLOLOW = 88001        Starting page of ASFDICT.ASFLOD area
GJFLOPSZ = 4276         Page size for area(s) in 'ASFDLOD'
GJFLONPG = 2000         Number of pages (blocks) in 'ASFDLOD'
```

```
*         The EMPDEMO SEGMENT parameters define the files and
*         areas that comprise the EMPDEMO segment.  This segment
*         is required to build the Employee Skills Network Demo
*         database.

GJEEMDSN = @DBPREFIX@.EMPDEMO.EMPDEMO
GJEEMDD  = EMPDEMO
GJEEMUNI = @UNIT@
GJEEMSER = @PACK@
GJEEMLOW = 75001        Starting page of EMPDEMO.EMPDEMO area
GJEEMPSZ = 4276         Page size for area(s) in 'EMPDEMO'
GJEEMNPG = 50           Number of pages (blocks) in 'EMPDEMO'

GJEINDSN = @DBPREFIX@.EMPDEMO.INSDEMO
GJEINDD  = INSDEMO
GJEINUNI = @UNIT@
GJEINSER = @PACK@
GJEINLOW = 75101        Starting page of INSDEMO area
GJEINPSZ = 4276         Page size for area(s) in 'INSDEMO'
GJEINNPG = 25           Number of pages (blocks) in 'INSDEMO'

GJEORDSN = @DBPREFIX@.EMPDEMO.ORGDEMO
GJEORDD  = ORGDEMO
GJEORUNI = @UNIT@
GJEORSER = @PACK@
GJEORLOW = 75151        Starting page of ORGDEMO area
GJEORPSZ = 4276         Page size for area(s) in 'ORGDEMO'
GJEORNPG = 25           Number of pages (blocks) in 'ORGDEMO'
```

```
*         The SQLDEMO SEGMENT parameters define the files and
*         areas that comprise the SQLDEMO segment.  This segment
*         is required to build the SQL version of the Employee
*         Skills Demo database.

GJQEMDSN = @DBPREFIX@.SQLDEMO.EMPLDEMO
GJQEMDD  = EMPLDEMO
GJQEMUNI = @UNIT@
GJQEMSER = @PACK@
GJQEMLOW = 77001        Starting page of SQLDEMO.EMPLDEMO area
GJQEMPSZ = 4276         Page size for area(s) in 'EMPLDEMO'
GJQEMNPG = 100          Number of pages (blocks) in 'EMPLDEMO'

GJQIFDSN = @DBPREFIX@.SQLDEMO.INFODEMO
GJQIFDD  = INFODEMO
GJQIFUNI = @UNIT@
GJQIFSER = @PACK@
GJQIFLOW = 77201        Starting page of SQLDEMO.INFODEMO area
GJQIFPSZ = 4276         Page size for area(s) in 'INFODEMO'
GJQIFNPG = 50           Number of pages (blocks) in 'INFODEMO'

GJQIXDSN = @DBPREFIX@.SQLDEMO.INDXDEMO
GJQIXDD  = INDXDEMO
GJQIXUNI = @UNIT@
GJQIXSER = @PACK@
GJQIXLOW = 77301        Starting page of SQLDEMO.INDXDEMO area
GJQIXPSZ = 4276         Page size for area(s) in 'INDXDEMO'
GJQIXNPG = 50           Number of pages (blocks) in 'INDXDEMO'

*         The following parameters define the PROJSEG segment.
*         This segment is part of the SQL demo database.

GJPRJDSN = @DBPREFIX@.PROJSEG.PROJDEMO
GJPRJDD  = PROJDEMO
GJPRJUNI = @UNIT@
GJPRJSER = @PACK@
GJPRJLOW = 77401        Starting page of PROJSEG.PROJDEMO area
GJPRJPSZ = 4276         Page size for area(s) in 'PROJDEMO'
GJPRJNPG = 50           Number of pages (blocks) in 'PROJDEMO'

*         The following parameters are used to define a SYSCTL
*         file.

GJCTLDSN = @DBPREFIX@.SYSCTL
GJCTLDD  = SYSCTL
GJCTLUNI = @UNIT@
GJCTLSER = @PACK@
GJCTLBLK = 20
```

```
*         The following parameters are used to define the
*         system disk and tape journals.  At least two journal
*         files are required.

GJDJ1DSN = @DBPREFIX@.J1JRNL
GJDJ1DD  = J1JRNL
GJDJ1UNI = @UNIT@
GJDJ1SER = @PACK@
GJDJ1PSZ = 2004         Page size for area(s) in 'J1JRNL'
GJDJ1NPG = 5000         Number of pages (blocks) in 'J1JRNL'

GJDJ2DSN = @DBPREFIX@.J2JRNL
GJDJ2DD  = J2JRNL
GJDJ2UNI = @UNIT@
GJDJ2SER = @PACK@
GJDJ2PSZ = 2004         Page size for area(s) in 'J2JRNL'
GJDJ2NPG = 5000         Number of pages (blocks) in 'J2JRNL'

GJDJ3DSN = @DBPREFIX@.J3JRNL
GJDJ3DD  = J3JRNL
GJDJ3UNI = @UNIT@
GJDJ3SER = @PACK@
GJDJ3PSZ = 2004         Page size for area(s) in 'J3JRNL'
GJDJ3NPG = 5000         Number of pages (blocks) in 'J3JRNL'

GJDJ4DSN = @DBPREFIX@.J4JRNL
GJDJ4DD  = J4JRNL
GJDJ4UNI = @UNIT@
GJDJ4SER = @PACK@
GJDJ4PSZ = 2004         Page size for area(s) in 'J4JRNL'
GJDJ4NPG = 5000         Number of pages (blocks) in 'J4JRNL'



*-------------------------------------------------------------------*
*                                                                   *
*    The following parameters are for an ADDON install.  If you are *
*    installing CA-IDMS/DB, you are not doing an ADDON install.     *
*                                                                   *
*    If you have already installed CA-IDMS/DB Release 15.0 and now  *
*    you are installing an additional product (for example, the     *
*    SQL option) into the same system, this is an ADDON install.    *
*                                                                   *
*-------------------------------------------------------------------*
GJNUMJRN =  ''          Number of disk journals defined to the CV
GJNWDMCL =  ''          Different name from your production global DMCL
```

C.2 CA-IDMS Tools VARBLIST Member Listing

This appendix contains a listing of CAISAG parameters. You can refer to this listing during the CA-IDMS Tools installation process.

```
GENLEVEL = F00103L8100
*************************************************************************
***    This is the list of variables that will be used for input   ***
***    by the SPECIFY AND GENERATE program.  Customize this        ***
***    member (VARBLIST), the PASSWORD member, the JOBCARD         ***
***    member and the SETUP member (if needed) before submitting   ***
***    the CAISAG job.                                             ***
***                                                                ***
***    The values in this list will be translated to uppercase     ***
***    unless they are enclosed in single quotes.                  ***
***                                                                ***
***    There are three valid types of lines in this list:          ***
***          a comment (asterisk in column 1)                      ***
***          a completely blank line                               ***
***          a variable line                                       ***
***                                                                ***
***    Variable lines consist of three fields:  a variable         ***
***    name, an equals sign, and a value.  If there are any        ***
***    blanks in the value you specify, enclose the value in       ***
***    single quotes.  Any characters following the value          ***
***    are ignored.                                                ***
***                                                                ***
***    For a quick check of your changes, enter the following      ***
***    three commands:                                             ***
***             EX ALL; F ALL P'<' 1 25; EX ALL '*' 1              ***
***                                                                ***
***    In the PASSWORD member, overtype the default passwords      ***
***    with the passwords supplied on your Product Authorization   ***
***    sheets.  Any unused password lines can be deleted.          ***
***                                                                ***
***                                                                ***
*************************************************************************



*************************************************************
**                                                         **
**     Do not change the order of the following variables.  **
**     These are the global substitution values.  The       **
**     values that you code here will be used as defaults   **
**     for other variables.                                 **
**                                                         **
*************************************************************
DSPREFIX = IDMS.R150    Dataset prefix for libraries
DBPREFIX = IDMS.TEST    Dataset prefix for database files
UNIT     = SYSDA        Unit used for permanent disk datasets
PACK     = IDMS01       Disk VOLSER used for permanent datasets
SBLK     = 8880         Blocksize used for source libraries
LBLK     = 18452        Blocksize used for load libraries
```

```
************************************************************
**                                                        **
**     Following are the products that can be installed   **
**     from this tape.  Specify either 'NO' or 'INSTALL'  **
**     for each product.                                  **
**                                                        **
**     If you have already done a CA-IDMS/TOOLS base      **
**     install and now you are installing an additional   **
**     product, that is an ADDON install.  For an ADDON   **
**     install, specify 'INSTALL' for the additional      **
**     product and 'NO' for all the other products.       **
**                                                        **
************************************************************
CA-IDMS/ADS-ALIVE                  = NO
CA-IDMS/ADS-TRACE                  = NO
CA-IDMS/DB-ANALYZER                = NO
CA-IDMS/DB-AUDIT                   = NO
CA-IDMS/DB-EXTRACTOR               = NO
CA-IDMS/DB-REORG                   = NO
CA-IDMS/DC-SORT                    = NO
CA-IDMS/DICTIONARY MIGRATOR        = NO
CA-IDMS/DICTIONARY MODULE EDITOR   = NO
CA-IDMS/DICTIONARY QUERY FACILITY  = NO
CA-IDMS/DML-ONLINE                 = NO
CA-IDMS/ENFORCER                   = NO
CA-IDMS/JOURNAL ANALYZER           = NO
CA-IDMS/LOG ANALYZER               = NO
CA-IDMS/MASTERKEY                  = NO
CA-IDMS/ONLINE LOG DISPLAY         = NO
CA-IDMS/SASO                       = NO
CA-IDMS/SCHEMA MAPPER              = NO
CA-IDMS/TASK ANALYZER              = NO


*    The next parameter determines whether the CICS option for
*    DC-SORT will be installed. If you specify 'YES', DC-SORT must
*    also be selected for installation or previously installed
*    at this genlevel.  Specify 'YES' or 'NO'.

WIO2CICS = NO           Install CA-IDMS/DC-SORT for CICS (YES/NO)


*    The next parameter determines whether the CICS option for
*    DML-ONLINE will be installed. If you specify 'YES', DML-ONLINE
*    must also be selected for installation or previously installed
*    at this genlevel.  Specify 'YES' or 'NO'.

WIKJCICS = NO           Install CA-IDMS/DMLO for CICS (YES/NO)
```

```
************************************************************
**                                                        **
**     The jobs which are generated will be output as     **
**     separate members.  The last two characters of the  **
**     member names will be ascending numbers.  You can   **
**     specify the beginning of the member names, up to 6 **
**     characters.  If you specify JOB, for example, the  **
**     member names will be JOB01, JOB02, JOB03 etc.      **
**                                                        **
************************************************************
BEGINMEM = JOB


************************************************************
**                                                        **
**     There are two ways that CA-IDMS/TOOLS Release 15.0 **
**     can be installed.  You can install a completely new **
**     15.0 system, keeping nothing from your 12.0 system, **
**     or you can install only the 15.0 software,         **
**     retaining your 12.0 dictionaries.  The latter is   **
**     called an upgrade.                                 **
**                                                        **
**     If converting from a release prior to 12.0, you may **
**     NOT do an upgrade.                                 **
**                                                        **
************************************************************
WIUPGRAD = NO          Specify 'NO' to do a complete install
*                      Specify 'YES' to do an upgrade from 12.0


************************************************************
**                                                        **
**     Specify your site's names for the following        **
**     datasets.                                          **
**                                                        **
************************************************************
GJCICDSN = CICS.LOADLIB Dataset name of the CICS load library --
*                            ignored if not installing DC-SORT for
*                            CICS or DML-ONLINE for CICS

PYCSIDSN = @DSPREFIX@.CSI              CA-IDMS 15.0 SMP/E CSI
PYMTSDSN = @DSPREFIX@.SMPMTS           CA-IDMS 15.0 SMP/E SMPMTS library
GJLLDDSN = @DSPREFIX@.DISTLOAD         CA-IDMS 15.0 distribution loadlib
GJMLDDSN = @DSPREFIX@.DISTMAC          CA-IDMS 15.0 distribution maclib
GJSLDDSN = @DSPREFIX@.DISTSRC          CA-IDMS 15.0 distribution srclib
GJLLTDSN = @DSPREFIX@.LOADLIB          CA-IDMS 15.0 target loadlib
GJOLTDSN = @DSPREFIX@.PPOPTION         CA-IDMS 15.0 PPOPTION library
GJAPFDSN = @DSPREFIX@.APFLIB           CA-IDMS 15.0 target loadlib for SVC
GJDBADSN = @DSPREFIX@.DBA.LOADLIB      CA-IDMS 15.0 DBA loadlib
GJCDSN   = CAC.LOADLIB                 CA-C runtime loadlib (CAILIB)
GJMACDSN = SYS1.MACLIB                 MVS system macro library
GJMKJDSN = SYS1.MODGEN                 2ns MVS system macro library
```

```
GJSTEPL1 = ''           First additional user steplib dataset
GJSTEPL2 = ''           Second additional user steplib dataset

GJSDMDSN = @DBPREFIX@.SYSTEM.DDLDML               SYSTEM.DDLDML
GJSLODSN = @DBPREFIX@.SYSTEM.DDLDCLOD             SYSTEM.DDLDCLOD
GJMSGDSN = @DBPREFIX@.SYSMSG.DDLDCMSG             SYSMSG.DDLDCMSG
GJCCTDSN = @DBPREFIX@.CATSYS.DCCAT                CATSYS.DCCAT
GJCCLDSN = @DBPREFIX@.CATSYS.DCCATLOD             CATSYS.DCCATLOD
GJCCXDSN = @DBPREFIX@.CATSYS.DCCATX               CATSYS.DCCATX


************************************************************
**                                                        **
**     Specify the variables below to describe your site.   **
**     These values will be used in the generated JCL.      **
**                                                        **
************************************************************
GJSYSOUT = *            System output class
GJWRKUNI = SYSDA        Disk unit for work files
GJTAPUNI = TAPE         Tape unit
TAPCLASS = T            Job class for tape jobs
GJLNKNAM = IEWL         Program name of linkage editor or binder
WIOSLNKP = 'LET,LIST,SIZE=(524288,65536),NCAL'   Linkedit parms
GJDJ1DD  = J1JRNL       DDNAME for first disk journal file
GJDJ2DD  = J2JRNL       DDNAME for second disk journal file
GJDJ3DD  = J3JRNL       DDNAME for third disk journal file (ddname or '')
GJDJ4DD  = J4JRNL       DDNAME for fourth disk journal file (ddname or '')
GJTJRDD  = SYSJRNL      DDNAME for tape journal file
WICVMODE = YES          Run DBTBLE, DMCL and SYSGEN under CV (YES/NO)
GJCTLDD  = SYSCTL       Default ddname for SYSCTL file
GJCTLDSN = @DBPREFIX@.SYSCTL Dsname of SYSCTL file
GJDCSYS  = 120          CA-IDMS system version number
WIAUTH   = ''           Authorized userid for SYSTEM.DDLDML area
WIAUTHPW = ''           Authorized password for dictionary signon
GJGLDMCL = R150DMCL     Your existing CA-IDMS global DMCL
WIDBTABL = R150DBTB     Database name table
WINWDMCL = NEWDMCL      Name for new global DMCL that will be created.
*                       Specify a name different from your production
*                       global DMCL.
GJSVCNUM = 174          The SVC number for CA-IDMS
WIBUFTYP = IDMS         Storage type of DMCL buffers (IDMS/OPSYS)

GJCASE   = MIXED        This variable is for European sites. Specify
*                       MIXED unless your system has terminals which
*                       cannot print mixed case characters.  (MIXED/UPPER)

WIGJCICS = NO           Is CA-IDMS/CICS installed (YES/NO)
WICTPNAM = CICS         Tpname specified in IDMSINTC

WIGJPRPK = NO           Is CA-IDMS/PRESS PACK installed (YES/NO)

WIADS    = YES          Is CA-ADS installed (YES/NO)
```

```
*--------------------------------------*
* SYSTEM DICTIONARY PARAMETERS         *
*--------------------------------------*
GJSDMDD  = DCDML        Default ddname for DCDML
GJSLODD  = DCLOD        Default ddname for DCLOD
GJMSGDD  = DCMSG        Default ddname for DCMSG

GJCCTDD  = DCCAT        Default ddname for DCCAT
GJCCLDD  = DCCATL       Default ddname for DCCATL
GJCCXDD  = DCCATX       Default ddname for DCCATX

*--------------------------------------*
* SYSTEM DEFAULT DICTIONARY PARAMETERS  *
*--------------------------------------*
*   Use the following parameters to specify the default dictionary
*   for your CA-IDMS environment.  These parameters default to the
*   system dictionary parameters.
WIDDMNAM = SYSTEM       DBname for default dictionary
WIDDMDD  = DCDML        DDname for default dictionary DDLDML
WIDLODD  = DCLOD        DDname for default dictionary DDLDCLOD
WIDDMDSN = @DBPREFIX@.SYSTEM.DDLDML      Dsname for default DDLDML
WIDLODSN = @DBPREFIX@.SYSTEM.DDLDCLOD    Dsname for default DDLDCLOD


************************************************************
**                                                        **
**     The following variables describe the CA-IDMS/TOOLS   **
**     dictionary.  This is the dictionary into which the   **
**     dialogs, work records, schemas, subschemas, maps,    **
**     applications and online tutorial modules required    **
**     for the CA-IDMS/TOOLS will be installed.             **
**                                                        **
**     You do not need to define a new dictionary.  You     **
**     can add the Tools dialogs, etc. into an application  **
**     dictionary by specifying WITDMNEW=NO and specifying  **
**     the name of your application dictionary in the       **
**     WITDMNAM variable.                                   **
**                                                        **
**     For an UPGRADE install, your CA-IDMS/TOOLS           **
**     dictionary already exists.  Specify WITDMNEW=NO      **
**     and specify the name of the dictionary in the        **
**     WITDMNAM variable.                                   **
**                                                        **
************************************************************

WITDMNEW = YES          Allocate CA-IDMS/TOOLS dictionary (YES/NO)
*                       YES:  a new dictionary will be allocated
*                             using the following parameters
*                       NO:   the following parameters identify
*                             an existing dictionary
```

```
*    If the access method is BDAM, the WIxxxVSP, WIxxxVCT and
*    WIxxxCSZ variables are ignored.

* TOOLDICT DDLDML AREA

WITDMNAM = TOOLDICT     DBname for TOOLDICT.DDLDML
WITDMDD  = TDICTDB      Default ddname for TDICTDB
WITDMDSN = @DBPREFIX@.TOOLDICT.DDLDML    Dsname for TOOLDICT.DDLDML
WITDMUNI = @UNIT@       Disk unit for TOOLDICT.DDLDML
WITDMSER = @PACK@       Disk VOLSER for TOOLDICT.DDLDML
WITDMDEV = BDAM         Access method
WITDMVSP = ''           Vsam space
WITDMVCT = DEFAULT      Vsam catalog
WITDMCSZ = 4096         CI size
WITDMPSZ = 4276         Pagesize (blocksize) of area in TDICTDB
WITDMNPG = 2000         Number of pages for  TOOLDICT.DDLDML
WITDMLPG = 95001        Starting page of TOOLDICT.DDLDML area
*
* TOOLDICT DDLDCLOD AREA
*
WITLODSN = @DBPREFIX@.TOOLDICT.DDLDCLOD Dsname of TOOLDICT.DDLDCLOD
WITLODD  = TDLODDB      Default ddname for TDLODDB
WITLOUNI = @UNIT@       Disk unit for TOOLDICT.DDLDCLOD
WITLOSER = @PACK@       Disk VOLSER for TOOLDICT.DDLDCLOD
WITLODEV = BDAM         Access method
WITLOVSP = ''           Vsam space
WITLOVCT = DEFAULT      Vsam catalog
WITLOCSZ = 4096         CI size
WITLOPSZ = 4276         Pagesize (blocksize) of area in TDLODDB
WITLONPG = 100          Number of pages (blocks) in TDLODDB
WITLOLPG = 98001        Start page of TOOLDICT.DDLDCLOD area


************************************************************
**                                                        **
**   The following parameter, WIGSISVC, should be changed **
**   to YES if you are installing either CA-IDMS/LOG      **
**   ANALYZER or CA-IDMS/TASK ANALYZER.  It modifies the  **
**   CA-IDMS SVC to access job accounting data and copy   **
**   it into the ERE.  The billing reports of both TASKA  **
**   and LOGA are dependent upon this ERE data.  If you   **
**   don't have the data gathered by specifying WIGSISVC  **
**   =YES, then the reports will not contain valid billing**
**   data for external run-units.                         **
**                                                        **
************************************************************
WIGSISVC = NO           Link GSISVCX into the CA-IDMS SVC (YES/NO)
```

```
************************************************************
**                                                        **
**     The following parameters describe the libraries    **
**     that will be allocated for this install.  If the   **
**     WI...PRI parameter is zero, the library will not be **
**     allocated and the parameters will be used to       **
**     identify an existing library.                      **
**                                                        **
************************************************************

* Indirect Libraries

WIINSDSN = @DSPREFIX@.INDTSRC
WIINSUNI = @UNIT@       Disk unit
WIINSSER = @PACK@       Disk VOLSER
WIINSSPC = CYL          Space unit
WIINSPRI = 25           Primary
WIINSSEC = 1            Secondary
WIINSDIR = 25           Directory blocks
WIINSBLK = @SBLK@

WIINODSN = @DSPREFIX@.INDTOBJ
WIINOUNI = @UNIT@       Disk unit
WIINOSER = @PACK@       Disk VOLSER
WIINOSPC = CYL          Space unit
WIINOPRI = 40           Primary
WIINOSEC = 2            Secondary
WIINODIR = 75           Directory blocks
WIINOBLK = 3120

* Target Library

WILLTDSN = @DSPREFIX@.CWIF0LLT
WILLTUNI = @UNIT@       Disk unit
WILLTSER = @PACK@       Disk VOLSER
WILLTSPC = CYL          Space unit
WILLTPRI = 75           Primary
WILLTSEC = 2            Secondary
WILLTDIR = 200          Directory blocks
WILLTBLK = @LBLK@       Blksize

* Distribution Libraries

WISLDDSN = @DSPREFIX@.CWIF0SLD
WISLDUNI = @UNIT@       Disk unit
WISLDSER = @PACK@       Disk VOLSER
WISLDSPC = CYL          Space unit
WISLDPRI = 15           Primary
WISLDSEC = 1            Secondary
WISLDDIR = 25           Directory blocks
WISLDBLK = @SBLK@       Blksize
```

```
WIMLDDSN = @DSPREFIX@.CWIF0MLD
WIMLDUNI = @UNIT@       Disk unit
WIMLDSER = @PACK@       Disk VOLSER
WIMLDSPC = CYL          Space unit
WIMLDPRI = 2            Primary
WIMLDSEC = 1            Secondary
WIMLDDIR = 25           Directory blocks
WIMLDBLK = @SBLK@       Blksize

WILLDDSN = @DSPREFIX@.CWIF0LLD
WILLDUNI = @UNIT@       Disk unit
WILLDSER = @PACK@       Disk VOLSER
WILLDSPC = CYL          Space unit
WILLDPRI = 32           Primary
WILLDSEC = 2            Secondary
WILLDDIR = 200          Directory blocks
WILLDBLK = @LBLK@       Blksize


************************************************************
**                                                        **
**     The following sets of parameters are specific to   **
**     particular products.  If you are not installing the **
**     product, the parameters for that product are       **
**     ignored.                                           **
**                                                        **
**     For database files, the WIxxxNPG parameter         **
**     determines whether the database file is allocated  **
**     or whether the parameters are used to identify an  **
**     existing database file.                            **
**                                                        **
**     For database files, the WIxxxVSP, WIxxxCSZ and     **
**     WIxxxVCT parameters are ignored unless the access  **
**     method is VSAM.                                    **
**                                                        **
************************************************************


************************************
**COMMON PRODUCT RUN-TIME PARAMETERS *
************************************
WIHLPDIC = TOOLDICT     Dbname of dictionary for tools help module
WIHLPNOD = ''           Alternate node used for tools help modules
WIHLPVER = 1            Version number of tools help modules
**
**
***********************************
**GENERAL SORT RUN-TIME PARAMETERS *
***********************************
WIKPPV01 = 10000        Amount of in-core sort storage (0 thru N)
WIKPPV02 = 10000        Amount of secondary sort storage (0 thru N)
WIKPPV03 = 100          Number of records per page (0 thru N)
WIKPPV04 = N            Prevent users from SETLIMIT facility (Y/N)
WIKPPV05 = PF24         ADS preprocessor exit key (PA1...PF24)
```

```
*********************************
**ADS-ALIVE RUN-TIME PARAMETERS *
*********************************
WIO5PV01 = ADSALIVE     Task code to invoke ADS-ALIVE
WIO5PV02 = 3800         Offset for implant (do not change)
WIO5PV03 = Y            Area sweep for dialog wild cards (Y/N)
WIO5PV04 = Y            Non-interrupt mode allowed (Y/N)
WIO5PV05 = 3            Number of days to retain debug queue records
WIO5PV06 = 10           Number of days to retain profile queue record
WIO5PV07 = D            Default dictname or dictname in profile (D/P)


*****************************************
**DATABASE EXTRACTOR DATABASE PARAMETERS *
*****************************************
WIDUSDSN = @DBPREFIX@.DBX.USVFIL1
WIDUSDD  = USVFIL1              Default ddname
WIDUSUNI = @UNIT@               Disk unit
WIDUSSER = @PACK@               Disk VOLSER
WIDUSDEV = BDAM         Access method
WIDUSVSP = ''           Vsam space
WIDUSVCT = DEFAULT      Vsam catalog
WIDUSCSZ = 4096         CI size
WIDUSPSZ = 3476                 Page size
WIDUSNPG = 3000                 Number of pages
WIDUSLPG = 370000               Starting page


*****************************************
**DATABASE EXTRACTOR RUN-TIME PARAMETERS *
*****************************************
WIOTPV01 = DBX          Task code to invoke DBX
WIOTPV02 = 50           Number of entries in DBX stack (30 to 1000)
WIOTPV03 = ANYONE       Who user can copy from (ANYONE/DBXADMIN/USER)
WIOTPV04 = YES          Retain physical seq of member records (YES/NO)
WIOTPV05 = YES          Extract owners of recursive records (YES/NO)
WIOTPV06 = YES          Begin VIEW/EDIT in middle of path (YES/NO)
WIOTPV07 = WARNING      How should NLYZ008 msg be issued (WARNING/ERROR)
**
**
*****************************************
**DC-SORT FOR CICS INSTALLATION PARAMETERS *
*****************************************
**
** There is a separate loadlib for DC-SORT for CICS.  This
** library should be a new library that contains only DC-SORT
** for CICS modules.  The library must be added to your
** CICS environment.
**
WIDLLDSN = @DSPREFIX@.DCSORT.LOADLIB
WIDLLDD  = DCSRLOAD     Default ddname
WIDLLUNI = @UNIT@       Disk unit
WIDLLSER = @PACK@       Disk VOLSER
WIDLLSPC = CYL          Space unit
WIDLLPRI = 1            Primary
WIDLLSEC = 1            Secondary
WIDLLDIR = 25           Directory blocks
WIDLLBLK = @LBLK@       Blksize
```

```
*****************************************
**DICTIONARY MIGRATOR RUN-TIME PARAMETERS *
*****************************************
WIMVPV01 = N            Exclude picture overrides
WIMVPV02 = N            Exclude subordinate element
WIMVPV03 = N            Map decompile option
WIMVPV04 = N            Ready in shared update (object dict)
WIMVPV05 = N            Ready in exclusive update   "     "
WIMVPV06 = N            Default is off
WIMVPV07 = N            Display programs with all relationships
WIMVPV08 = N            Exclude clist (DCMT V NEW COPIES)
WIMVPV09 = N            Exclude immediate in clist
WIMVPV10 = N            Exclude dictionary name in clist
WIMVPV11 = N            Exclude version in clist
WIMVPV12 = N            No UDC comment syntax
WIMVPV13 = N            Exclude all UDN references
WIMVPV14 = N            Exclude element UDN references
WIMVPV15 = N            Exclude attribute UDN references
WIMVPV16 = N            Exclude system UDN references
WIMVPV17 = N            Exclude record UDN references
WIMVPV18 = N            Exclude module UDN references
WIMVPV19 = N            Exclude program UDN references
WIMVPV20 = N            Exclude user UDN references
WIMVPV21 = N            To use double quote (") throughout IDD syntax
WIMVPV22 = N            Extract IDMSNTWK components
WIMVPV23 = N            Omit elements from extraction
WIMVPV24 = N            Omit elements from extract when COBOLFMT is used
WIMVPV25 = N            Extract system records
WIMVPV26 = N            Skip all UDN extraction
WIMVPV27 = N            Skip element UDN extraction
WIMVPV28 = N            Skip record UDN extraction
WIMVPV29 = N            Skip module UDN extraction
WIMVPV30 = N            Skip user UDN extraction
WIMVPV31 = N            Skip attribute UDN extraction
WIMVPV32 = N            Skip system UDN extraction
WIMVPV33 = N            Omit syntax file display report when RUN=IMPORT
WIMVPV34 = N            Use DELETE/ADD syntax (not MODIFY)
WIMVPV35 = N            Extract same as entities
WIMVPV36 = N            Abend on database error
WIMVPV37 = N            Omit extraction of class-attributes
WIMVPV38 = N            Omit extraction of class
WIMVPV39 = N            Omit extraction of systems
WIMVPV40 = N            Stop after validation if errors found (cc=8)
WIMVPV41 = N            Do not explode attribute network if LEVEL=ONLY
WIMVPV42 = N            Bypass source dictionary security checking
WIMVPV43 = N            Bypass target dictionary security checking
WIMVPV44 = N            Put entity type labels in the DDDLUPD file
WIMVPV45 = N            Create ADSOBCOM source gen keyfile
WIMVPV46 = N            Omit signon from syntax files (XEQUDAT parm)
WIMVPV47 = N            Omit signon from map syntax files RHDCDEL/RHDCUPD
WIMVPV48 = N            Skip extraction of entities with equal dates
**
```

```
**
*****************************************************
**DICTIONARY MIGRATOR ASSISTANT DATABASE PARAMETERS *
*****************************************************
WIDXDDSN = @DBPREFIX@.DMA.XDMFIL1
WIDXDDD  = XDMFIL1
WIDXDUNI = @UNIT@
WIDXDSER = @PACK@
WIDXDDEV = BDAM          Access method
WIDXDVSP = ''            Vsam space
WIDXDVCT = DEFAULT       Vsam catalog
WIDXDCSZ = 4096          CI size
WIDXDPSZ = 3476
WIDXDNPG = 600
WIDXDLPG = 0300000


*****************************************************
**DICTIONARY MIGRATOR ASSISTANT RUN-TIME PARAMETERS *
*****************************************************
WIMUPTSK = DMA           Task code to invoke DMA


***********************************************
**DICTIONARY MODULE EDITOR RUN-TIME PARAMETERS *
***********************************************
WIM3PV01 = Y             Set longterm dbkey locks (Y/N)
WIM3PV02 = PAGE          Scroll amount (PAGE/HALF/CSR)
WIM3PV03 = ;             Command delimiter
WIM3PV04 = N             Padding character, nulls or blanks (N,B)
WIM3PV05 = HIGHEST       Default IDD version (HIGHEST/LOWEST)
WIM3PV06 = I             Security system (I=IDD, D=DBMS, B=DBMS+IDD)
WIM3PV07 = INPUT         Userid changes (INPUT=ALLOW, PROT=NOT ALLOW)
WIM3PV08 = Y             Module sort performed (Y/N)
WIM3PV09 = Y             Reset database/node to DME entry value (Y/N)
WIM3PV10 = Y             Clear key function (Y=END, N=RESHOW)


*********************************
**DML-ONLINE DATABASE PARAMETERS *
*********************************
WIDPRDSN = @DBPREFIX@.DMLO.PROFILE
WIDPRDD  = USDFIL1
WIDPRUNI = @UNIT@
WIDPRSER = @PACK@
WIDPRDEV = BDAM
WIDPRVSP = ''
WIDPRVCT = DEFAULT
WIDPRCSZ = 4096
WIDPRPSZ = 3476
WIDPRNPG = 600
WIDPRLPG = 0360000
```

```
*********************************
**DML-ONLINE RUNTIME PARAMETERS *
*********************************
WIKJPV01 = N            Lower case option (Y/N)
WIKJPV02 = 1            IDMS DC print class
WIKJPV33 = A            TSO print class
WIKJPV34 = A            CICS print class
WIKJPV03 = COBOL        Display format (COBOL/vertical)
WIKJPV04 = ON           AUTOHEX option (ON/OFF)
WIKJPV05 = ON           AUTO-BIND option (ON/OFF)
WIKJPV06 = FAST         Data updates and command/PFkey input (FAST/STEP)
WIKJPV07 = FAST         Clist execution  (FAST/STEP)
WIKJPV08 = INPUT        Command line echo display (INPUT/USED)
WIKJPV09 = NORM         LRF screen format  (NORM/MAX)
WIKJPV10 = EXPERT       Session mode  (EXPERT/MENU)
WIKJPV11 = (DYNAM,OFF)  User exit option
WIKJPV12 = DMLOSYS      Owner ID for global (system owned) profile/CLIST
WIKJPV13 = USERID01     DMLO administrator signon
WIKJPV14 = USERID02     DMLO administrator signon
WIKJPV15 = INPUT        Chg IDMS/DC userid at DMLO signon? (INPUT/PROT)
WIKJPV16 = C'_'         Nondisplay translation
WIKJPV17 = ''           Default signon dictionary
WIKJPV18 = ''           Default signon dictionary node
WIKJPV19 = DMLO         Profile segment (DB) name
WIKJPV20 = ''           Profile segment (DB) node
WIKJPV21 = SBUF         Default scratch record name prefix
WIKJPV22 = QBUF         Default queue record name prefix
WIKJPV23 = 4096         Default scratch/queue buffer length
WIKJPV24 = PA1          Attention/interrupt
WIKJPV25 = (PF2,PF14)   Signon help pfkeys
WIKJPV26 = (PF4,PF16)   Profile list pfkeys
WIKJPV27 = (PF1,PF13)   Session help pfkeys
WIKJPV28 = (PF2,PF14)   Show pfkeys
WIKJPV29 = (PF3,PF15)   Pfkeys for END
WIKJPV30 = (PF4,PF16)   Pfkeys used to request redisplay function
WIKJPV31 = (PF7,PF19)   Pfkeys to scroll up
WIKJPV32 = (PF8,PF20)   Pfkeys to scroll down
WIKJPV35 = Y            Default use of ENTER key (Y/N)
*                          Y:  Do not re-execute command
*                          N:  Re-execute command
WIKJPV36 = ''           Default signon screen DBNAME
*                          A value specified here will appear each time
*                          the DMLO signon screen is presented.


*********************************
**ENFORCER DATABASE PARAMETERS *
*********************************
WIECTDSN = @DBPREFIX@.ENFORCER.CTRL
WIECTDD  = ESXFIL1      Ddname
WIECTUNI = @UNIT@       Disk unit
WIECTSER = @PACK@       Disk VOLSER
WIECTDEV = BDAM         Access method
WIECTVSP = ''           VSAM space
WIECTVCT = DEFAULT      VSAM catalog
WIECTCSZ = 4096         CI size
WIECTPSZ = 9076         Page size
WIECTNPG = 500          Number of pages
WIECTLPG = 0310001      Low page
```

```
WIELODSN = @DBPREFIX@.ENFORCER.LOAD
WIELODD  = ESXFIL2      Ddname
WIELOUNI = @UNIT@       Disk unit
WIELOSER = @PACK@       Disk VOLSER
WIELODEV = BDAM         Access method
WIELOVSP = ''           VSAM space
WIELOVCT = DEFAULT      VSAM catalog
WIELOCSZ = 4096         CI size
WIELOPSZ = 9076         Page size
WIELONPG = 200          Number of pages
WIELOLPG = 0310751      Low page

WIEINDSN = @DBPREFIX@.ENFORCER.INDEX
WIEINDD  = ESXFIL3      Ddname
WIEINUNI = @UNIT@       Disk unit
WIEINSER = @PACK@       Disk VOLSER
WIEINDEV = BDAM         Access method
WIEINVSP = ''           VSAM space
WIEINVCT = DEFAULT      VSAM catalog
WIEINCSZ = 4096         CI size
WIEINPSZ = 9076         Page size
WIEINNPG = 100          Number of pages
WIEINLPG = 0311001      Low page


***********************************
**ENFORCER LOAD LIBRARY PARAMETERS *
***********************************

*  There is a separate loadlib for Enforcer into which IDMS
*  utilities will be relinked to interface with Enforcer.
*  This library should be a new library that will contain only
*  these relinked modules.

WIELLDSN = @DSPREFIX@.ENFORCER.LOADLIB
WIELLDD  = ENFRLOAD
WIELLUNI = @UNIT@
WIELLSER = @PACK@
WIELLSPC = CYL          Space unit
WIELLPRI = 1            Primary
WIELLSEC = 1            Secondary
WIELLDIR = 25           Directory blocks
WIELLBLK = @LBLK@       Blksize

*******************************
**ENFORCER RUN-TIME PARAMETERS *
*******************************
WIO3PV01 = ENFORCER     Task code to invoke ENFORCER
WIO3PV02 = D            Deadlock process (D-DEADLOCK, B-BATCH, M=IDDM)
WIO3PV03 = Y            Space delimited words allowed (Y=YES, N=NO)
WIO3PV04 = Y            Dash(-) delimited words allowed (Y=YES, N=NO)
WIO3PV05 = Y            Uline(_) delimited word allowed (Y=YES, N=NO)
```

```
*********************************
**MASTERKEY DATABASE PARAMETERS *
*********************************
WIMDADSN = @DBPREFIX@.MASTRKEY.DATASEG
WIMDADD  = SSKFIL1      Ddname
WIMDAUNI = @UNIT@       Disk unit
WIMDASER = @PACK@       Disk VOLSER
WIMDADEV = BDAM         Access method
WIMDAVSP = ''           VSAM space
WIMDAVCT = DEFAULT      VSAM catalog
WIMDACSZ = 4096         CI size
WIMDAPSZ = 3476         Page size
WIMDANPG = 600          Number of pages
WIMDALPG = 0330000      Low page

*********************************
**MASTERKEY RUN-TIME PARAMETERS *
*********************************
WIF8PV01 = TOOLDICT     Dictname for transient clists
WIF8PV02 = ''           Dictnode for transient clists


*****************************************
**ONLINE LOG DISPLAY RUN-TIME PARAMETERS *
*****************************************
WIF9PTSK = LOGD         Task code to invoke online log display


*********************************************
**SASO DEFAULT DOCUMENT DATABASE PARAMETERS *
*********************************************
WISPRDSN = @DBPREFIX@.SASO.PRIMARY
WISPRDD  = ESSFIL1      Ddname
WISPRUNI = @UNIT@       Disk unit
WISPRSER = @PACK@       Disk VOLSER
WISPRDEV = BDAM         Access method
WISPRVSP = ''           VSAM space
WISPRVCT = DEFAULT      VSAM catalog
WISPRCSZ = 4096         CI size
WISPRPSZ = 9076         Page/block size
WISPRNP1 = 95           Number of pages/blocks for ESS-CTRL-AREA
WISPRLP1 = 8100001      Start page for ESS-CTRL-AREA
WISPRLB1 = 1            Start block for ESS-CTRL-AREA
WISPRNP2 = 760          Number of pages/blocks for ESS-TEXT-AREA
WISPRLP2 = 8100101      Start page for ESS-TEXT-AREA
WISPRLB2 = 96           Start block for ESS-TEXT-AREA
WISPRNP3 = 95           Number of pages/blocks for ESS-INDEX-AREA
WISPRLP3 = 8100901      Start page for ESS-INDEX-AREA
WISPRLB3 = 856          Start block for ESS-INDEX-AREA
```

```
WISREDSN = @DBPREFIX@.SASO.RELEASE
WISREDD  = ESSFIL2      Ddname
WISREUNI = @UNIT@       Disk unit
WISRESER = @PACK@       Disk VOLSER
WISREDEV = BDAM         Access method
WISREVSP = ''           VSAM space
WISREVCT = DEFAULT      VSAM catalog
WISRECSZ = 4096         CI size
WISREPSZ = 9076         Page size
WISRENP1 = 240          Number of pages/blocks for ESS-RELSE-AREA
WISRELP1 = 8101001      Start page for ESS-RELSE-AREA
WISRELB1 = 1            Start block for ESS-RELSE-AREA
WISRENP2 = 235          Number of pages/blocks for ESS-RTEXT-AREA
WISRELP2 = 8101301      Start page for ESS-RTEXT-AREA
WISRELB2 = 241          Start block for ESS-RTEXT-AREA

WISDODSN = @DBPREFIX@.SASO.DOCUMENT
WISDODD  = ESSFIL3      Ddname
WISDOUNI = @UNIT@       Disk unit
WISDOSER = @PACK@       Disk VOLSER
WISDODEV = BDAM         Access method
WISDOVSP = ''           VSAM space
WISDOVCT = DEFAULT      VSAM catalog
WISDOCSZ = 4096         CI size
WISDOPSZ = 9076         Block/page size for SASO database
WISDONPG = 3            Number of pages/blocks for ESS-CTRLD-AREA
WISDOLPG = 8101601      Start page for ESS-CTRLD-AREA
WISDOLB1 = 1            Start block for ESS-CTRLD-AREA

WISSPDSN = @DBPREFIX@.SASO.SPGTEXT
WISSPUNI = @UNIT@       Disk unit
WISSPSER = @PACK@       Disk VOLSER
WISASOID = 254          Unique internal id for SASO doc (36 TO 255)
```

```
***************************
**SASO RUN-TIME PARAMETERS *
***************************
WIO4PV01 = SASO         Task code to invoke SASO
WIO4PV02 = SPG          Default document DBNAME table entry

*  The following parameters are the default JCL jobcard lines
*  for initial SASO user profiles.
*  NOTE: JCL values can be 1 TO 48 characters long and must be
*        enclosed in single quotes

WIO4PV03 = ''           Default jobcard-1 for initial user profiles
WIO4PV04 = ''           Default jobcard-2 for initial user profiles
WIO4PV05 = ''           Default jobcard-3 for initial user profiles



************************************************************
**                                                        **
**    Specify whether this is an ADDON install.           **
**                                                        **
**    NO:   1.  This is the initial installation of the   **
**              Release 15.0 CA-IDMS/TOOLS on this system.**
**          2.  All general subproducts will be installed **
**              if one or more main products are selected.**
**          3.  The CA-IDMS/TOOLS dictionary is allocated **
**              and defined to the system if WIUPGRAD=NO. **
**          4.  The target and distribution libraries are **
**              allocated.                                **
**                                                        **
**    YES:  1.  You have previously installed some of the **
**              Release 15.0 CA-IDMS/TOOLS and are now    **
**              installing additional tools.              **
**          2.  The general subproducts will not be       **
**              installed because they are already       **
**              installed.                                **
**          3.  The CA-IDMS/TOOLS dictionary will not be  **
**              allocated, even if WIUPGRAD=NO.           **
**          4.  The target and distribution libraries will**
**              not be allocated because they are already **
**              allocated.                                **
**                                                        **
************************************************************
WIADDON  = NO           Additional product install (YES/NO)
```

C.3 CA-Endevor/DB VARBLIST Member Listing

```
GENLEVEL = F00103SJE00
**********************************************************************
**    This is the list of variables that will be used for input   **
**    by the SPECIFY AND GENERATE program.  Customize this        **
**    this member (VARBLIST), the PASSWORD member, the JOBCARD    **
**    member and the SETUP member (if needed) before submitting   **
**    the CAISAG job.                                             **
**                                                                **
**    The values in this list will be translated to uppercase     **
**    unless they are enclosed in single quotes.                  **
**                                                                **
**    There are three valid types of lines in this list:          **
**          a comment (asterisk in column 1)                      **
**          a completely blank line                               **
**          a variable line                                       **
**                                                                **
**    Variable lines consist of three fields:  a variable         **
**    name, an equals sign, and a value.  If there are any        **
**    blanks in the value you specify, enclose the value in       **
**    single quotes.  Any characters following the value          **
**    are ignored.                                                **
**                                                                **
**    For a quick check of your changes, enter the following      **
**    three commands:                                             **
**             EX ALL; F ALL P'<' 1 25; EX ALL '*' 1              **
**                                                                **
**    In the PASSWORD member, overtype the default password       **
**    with the password supplied on your Product Authorization    **
**    sheet.                                                      **
**                                                                **
**********************************************************************


************************************************************
**                                                        **
**    Following are the products that can be installed    **
**    from this tape. Specify either 'NO' or 'INSTALL'    **
**    for each product.                                   **
**                                                        **
************************************************************

CA-ENDEVOR/DB = INSTALL



************************************************************
**                                                        **
**    Before installing CA-Endevor/DB, you must already   **
**    have installed CA-IDMS (Release 12.0, Release 14.0 or **
**    Release 14.1).  CA-Endevor/DB is installed into the **
**    same CSI and SMP/E environment as CA-IDMS and creates **
**    additional target and distribution libraries.       **
**                                                        **
************************************************************

DSPREFIX = IDMS.R150    Dataset prefix for libraries
DBPREFIX = IDMS.TEST    Dataset prefix for database files
UNIT     = SYSDA        Unit used for permanent disk datasets
PACK     = IDMS01       Disk VOLSER used for permanent datasets
SBLK     = 8880         Blocksize used for source libraries
LBLK     = 18452        Blocksize used for load libraries
```

```
************************************************************
**                                                        **
**    The jobs which are generated will be output as      **
**    separate members.  The last two characters of the   **
**    member names will be ascending numbers.  You can    **
**    specify the beginning of the member names, up to 6  **
**    characters.  If you specify JOB, for example, the   **
**    member names will be JOB01, JOB02, JOB03 etc.       **
**                                                        **
************************************************************

BEGINMEM = JOB


************************************************************
**                                                        **
**    The following parameter allows you to reinstall the **
**    same version of CA-Endevor/DB into the new CA-IDMS  **
**    CSI when you have installed a new release of        **
**    CA-IDMS.                                            **
**                                                        **
************************************************************

REINSTALL= NO


************************************************************
**                                                        **
**    Specify YES for the following parameter if you      **
**    have already installed an earlier release of        **
**    CA-Endevor/DB, have existing CCDBs and do not       **
**    want to allocate new database files.                **
**                                                        **
**    You cannot do an upgrade of CA-Endevor/DB into      **
**    any CSI where CA-Endevor/DB is already installed.   **
**                                                        **
************************************************************

ABUPGRAD = NO           Use the database files from the prior release? (YES/NO)
FROMR14  = YES          Prior release was Release 14.0 or later? (YES/NO)
```

```
************************************************************
**                                                        **
**    Specify the variables below to describe your site.  **
**    These values will be used in the generated JCL.     **
**                                                        **
************************************************************

ABSYSOUT = *            System output class
ABWRKUNI = SYSDA        Disk unit for work files
ABWRKSER = IDMS01       Disk VOLSER for work files
ABTAPUNI = TAPE         Tape unit
TAPCLASS = T            Job class for tape jobs
ABLNKNAM = IEWL         Program name of linkage editor or binder
ABCVMODE = YES          Run DBTBLE, DMCL and SYSGEN under CV (YES/NO)
GJGLDMCL = R150DMCL     Existing global DMCL
ABNWDMCL = ''           Name for new global DMCL that will be created.
*                       Specify a name different from your production
*                       global DMCL.

GJBUFTYP = IDMS         Storage type of DMCL buffers (IDMS/OPSYS)


*    GJPDETYP determines whether modules which are eligible to
*    use null PDEs will be defined during sysgen processing
*    (maps, dialogs and tables).
*
*    GJPDETYPE determines whether CA-IDMS will allocate program
*    definition elements (PDEs) at runtime (dynamic) or at
*    system startup (static).
*
*    If DYNAMIC is chosen, program definitions for eligible
*    modules will not be added to the sysgen.  This will reduce
*    system startup storage and will allow the Central Version to
*    use null PDEs for eligible modules.  It allows PDEs for
*    maps, dialogs and tables to be loaded in XA storage.
*
*
*    If STATIC is chosen, all modules will be added to the sysgen
*    as PDEs, using more storage at startup.
*
*    For more information, refer to the CA-IDMS Installation and
*    Maintenance Guide, Chapter 4.
```

```
GJPDETYP = DYNAMIC      (DYNAMIC/STATIC)


*    In the sysgen, programs can be defined for storage protection.
*    The STORPROT variable determines whether PROGRAM statements
*    in the sysgen will have PROTECT or NOPROTECT.
*
*    If STORPROT=YES, then all PROGRAM statements will have PROTECT
*    and the CV SYSTEM statement will have NOPROTECT.  This is the
*    sysgen compiler default, and is a good choice for test systems.
*    This choice installs NDVRTASK members into the SYSTEM dictionary.
*
*    If STORPROT=NO, then all PROGRAM statements will have NOPROTECT
*    and the CV SYSTEM statement will have PROTECT.  This is a good
*    choice for a production system.  This choice installs DNODxxxx
*    members into the SYSTEM dictionary.
*
*    DLODxxxx members contain sysgen source using PROTECT on the
*    PROGRAM statements.  DNODxxxx members contain sysgen source
*    using NOPROTECT on the PROGRAM statements.
*
*    No matter which value you specify for the STORPROT variable
*    below, both DLODxxxx members and DNODxxxx members will be
*    installed by SMP/E into your DISTSRC library.  The value you
*    choose for the STORPROT variable below determines only which
*    set of members is installed into the SYSTEM dictionary.
*    However, at a later time, you can rerun the IDMSDDDL steps
*    from Job 8, changing DLODxxxx to DNODxxxx or vice versa.

STORPROT = YES          Use storage PROTECT for programs (DLOD members)


************************************************************
**                                                        **
**    The following eight parameters pertain to the       **
**    message library (ISPMLIB).                          **
**                                                        **
**    If you have already installed CA-ENDEVOR/MVS, then  **
**    you already have a message library (ISPMLIB) and    **
**    only its dataset name is required.  This install    **
**    will add new members to it.                         **
**                                                        **
**    If you have not already installed CA-ENDEVOR/MVS,   **
**    then this install will also allocate the message    **
**    library.                                            **
**                                                        **
************************************************************
ABOLDMSG = YES          Have you already installed CA-ENDEVOR/MVS? (YES/NO)
ABMSGDSN = @DSPREFIX@.ISPMLIB Specify the/a name for this library.
*                       If ABOLDMSG=YES then DISP=SHR will be generated
*                       If ABOLDMSG=NO then DISP=(NEW,CATLG) will be generated
ABMSGSER = @PACK@       Disk VOLSER (Ignored if ABOLDMSG=YES)
ABMSGSPC = TRK          Space unit (Ignored if ABOLDMSG=YES)
ABMSGPRI = 5            Primary space (Ignored if ABOLDMSG=YES)
ABMSGSEC = 5            Secondary SPACE (Ignored if ABOLDMSG=YES)
ABMSGDIR = 5            Directory blocks (Ignored if ABOLDMSG=YES)
ABMSGBLK = @SBLK@       Blksize (Ignored if ABOLDMSG=YES)
```

```
************************************************************
**                                                        **
**    The following three parameters are used to determine  **
**    how to rebuild the SMP/E DDDEF for the SYSLIB        **
**    concatenation.                                      **
**                                                        **
************************************************************

ABCICS   = NO           Have you installed CA-IDMS/CICS SUPPORT (YES/NO)
ABTOOLS  = NO           Have you installed CA-IDMS/TOOLS (YES/NO)
ABWI#REL = 15.0         If YES, which TOOLS release (12.0/14.0/14.1,15.0)


************************************************************
**                                                        **
**    Most sites do not use BLP processing and should not   **
**    change the following variables.                     **
**                                                        **
************************************************************

ABBLP    = NO           YES only if you need to use BLP processing
ABBLPHLQ = CAI          High level qualifier of tape file DSN for BLP


************************************************************
**                                                        **
**    Most sites do not need uppercase modules and should   **
**    NOT change the following variable. It is for sites    **
**    that have terminals that cannot display mixed-case    **
**    characters and for sites that use double-byte         **
**    character support (DBCS).                           **
**                                                        **
************************************************************

ABCASE   = MIXED        Type of HELP panels (MIXED/UPPER)


************************************************************
**                                                        **
**    These parameters specify the DBTABLE information      **
**    required to add the CA/Endevor/DB CCDB segment.       **
**    The files in this segment will be included with       **
**    the DBNAME in the DBTABLE specified by ABUSRDB.       **
**                                                        **
************************************************************

GJDBTABL = R150DBTB    DBTABLE name to be modified
ABUSRDB  = ''          DBNAME to which the CCDB segment will be added
```

```
************************************************************
**                                                        **
**    These parameters specify information about the      **
**    existing segments in your CA-IDMS system.           **
**                                                        **
************************************************************

ABSYSDBN = SYSTEM      DBNAME containing your system definition
ABAUTH   = ''           Authorized user for dictionary signon (optional)
ABAUTHPW = ''           Password for the authorized user (optional)
GJDCSYS  = 120         CA-IDMS/DC system number to be modified
ABDRLDBN = SYSDIRL     DBNAME containing your IDMSDIRL information



************************************************************
**                                                        **
**    Specify your site's names for the following         **
**    datasets.  These are the names of your existing     **
**    CA-IDMS datasets.                                   **
**                                                        **
************************************************************

PYCSIDSN = @DSPREFIX@.CSI             CA-IDMS SMP/E CSI
PYMTSDSN = @DSPREFIX@.SMPMTS          CA-IDMS SMP/E SMPMTS library
GJLLDDSN = @DSPREFIX@.DISTLOAD        CA-IDMS distribution loadlib
GJMLDDSN = @DSPREFIX@.DISTMAC         CA-IDMS distribution maclib
GJSLDDSN = @DSPREFIX@.DISTSRC         CA-IDMS distribution srclib
GJLLTDSN = @DSPREFIX@.LOADLIB         CA-IDMS target loadlib
GJOLTDSN = @DSPREFIX@.PPOPTION        CA-IDMS PPOPTION library
GJDBADSN = @DSPREFIX@.DBA.LOADLIB     CA-IDMS DBA loadlib
GJMACDSN = SYS1.MACLIB                 MVS system macro library

GJSTEPL1 = ''           First additional user steplib dataset (optional)
GJSTEPL2 = ''           Second additional user steplib dataset (optional)

GJSDMDSN = @DBPREFIX@.SYSTEM.DDLDML           SYSTEM.DDLDML
GJSLODSN = @DBPREFIX@.SYSTEM.DDLDCLOD         SYSTEM.DDLDCLOD
GJMSGDSN = @DBPREFIX@.SYSMSG.DDLDCMSG         SYSMSG.DDLDCMSG
GJCCTDSN = @DBPREFIX@.CATSYS.DCCAT            CATSYS.DCCAT
GJCCLDSN = @DBPREFIX@.CATSYS.DCCATLOD         CATSYS.DCCATLOD
GJCCXDSN = @DBPREFIX@.CATSYS.DCCATX           CATSYS.DCCATX
GJDDMDSN = @DBPREFIX@.SYSDIRL.DDLDML          SYSDIRL.DDLDML
GJDLODSN = @DBPREFIX@.SYSDIRL.DDLDCLOD        SYSDIRL.DDLDCLOD
GJCTLDSN = @DBPREFIX@.SYSCTL                  Dsname of SYSCTL file
```

```
************************************************************
**                                                        **
**    Specify your site's DDNAMEs for the following       **
**    database datasets.                                  **
**                                                        **
************************************************************

GJSDMDD  = DCDML        Default ddname for DCDML
GJSLODD  = DCLOD        Default ddname for DCLOD
GJMSGDD  = DCMSG        Default ddname for DCMSG
GJCCTDD  = DCCAT        Default ddname for DCCAT
GJCCLDD  = DCCATL       Default ddname for DCCATL
GJCCXDD  = DCCATX       Default ddname for DCCATX
GJDDMDD  = DIRLDB       Default ddname for DIRLDB
GJDLODD  = DIRLLOD      Default ddname for DIRLLOD
GJCTLDD  = SYSCTL       Default ddname for SYSCTL file
GJDJ1DD  = J1JRNL       Default ddname for first disk journal file
GJDJ2DD  = J2JRNL       Default ddname for second disk journal file
GJDJ3DD  = J3JRNL       Default ddname for 3rd disk journal file (ddname or '')
GJDJ4DD  = J4JRNL       Default ddname for 4th disk journal file (ddname or '')
GJTJRDD  = SYSJRNL      Default ddname for tape journal file


************************************************************
**                                                        **
**    The following parameters describe the libraries     **
**    that will be allocated for this install.  If the    **
**    AB...PRI parameter is zero, the library will not be **
**    allocated and the parameters will be used to        **
**    identify an existing library.                       **
**                                                        **
************************************************************

************************************************************
**    Indirect Libraries                                  **
************************************************************

ABINSDSN = @DSPREFIX@.INDSRC
ABINSUNI = @UNIT@       Disk unit
ABINSSER = @PACK@       Disk VOLSER
ABINSSPC = TRK          Space unit
ABINSPRI = 5            Primary
ABINSSEC = 1            Secondary
ABINSDIR = 25           Directory blocks
ABINSBLK = @SBLK@

ABINODSN = @DSPREFIX@.INDOBJ
ABINOUNI = @UNIT@       Disk unit
ABINOSER = @PACK@       Disk VOLSER
ABINOSPC = TRK          Space unit
ABINOPRI = 35           Primary
ABINOSEC = 5            Secondary
ABINODIR = 25           Directory blocks
ABINOBLK = 3120
```

```
************************************************************
**    New SMP/E Target Library for CA-Endevor/DB          **
************************************************************

ABLLTDSN = @DSPREFIX@.CABF0LLT
ABLLTUNI = @UNIT@       Disk unit
ABLLTSER = @PACK@       Disk VOLSER
ABLLTSPC = TRK          Space unit
ABLLTPRI = 65           Primary
ABLLTSEC = 5            Secondary
ABLLTDIR = 35           Directory blocks
ABLLTBLK = @LBLK@       Blksize

************************************************************
**    New SMP/E Distribution Libraries for CA-Endevor/DB  **
************************************************************

ABSLDDSN = @DSPREFIX@.CABF0SLD
ABSLDUNI = @UNIT@       Disk unit
ABSLDSER = @PACK@       Disk VOLSER
ABSLDSPC = TRK          Space unit
ABSLDPRI = 5            Primary
ABSLDSEC = 1            Secondary
ABSLDDIR = 25           Directory blocks
ABSLDBLK = @SBLK@       Blksize

ABMLDDSN = @DSPREFIX@.CABF0MLD
ABMLDUNI = @UNIT@       Disk unit
ABMLDSER = @PACK@       Disk VOLSER
ABMLDSPC = TRK          Space unit
ABMLDPRI = 5            Primary
ABMLDSEC = 1            Secondary
ABMLDDIR = 5            Directory blocks
ABMLDBLK = @SBLK@       Blksize

ABLLDDSN = @DSPREFIX@.CABF0LLD
ABLLDUNI = @UNIT@       Disk unit
ABLLDSER = @PACK@       Disk VOLSER
ABLLDSPC = TRK          Space unit
ABLLDPRI = 90           Primary
ABLLDSEC = 5            Secondary
ABLLDDIR = 50           Directory blocks
ABLLDBLK = @LBLK@       Blksize
```

```
*************************************************************
**                                                         **
**    The CCDB SEGMENT parameters define the segment,      **
**    files and areas that comprise the CA-Endevor/DB      **
**    segment. The files in this segment will be included  **
**    with the DBNAME in the DBTABLE specified by the      **
**    ABUSRDB variable.                                    **
**                                                         **
*************************************************************

ABCCDSEG = CCDB         Segment name for CA-Endevor/DB CCDB

ABADMDSN = @DBPREFIX@.CCDB.CCDBADM   Dsname for 'NDVR-ADM' file
ABADMDD  = CCDBADM      File name (DD name) for 'NDVR-ADM' file
ABADMUNI = @UNIT@       Physical unit type (e.g. 3390)
ABADMSER = @PACK@       Physical disk volume for 'NDVR-ADM' file
ABADMLOW = 900001       Starting page for 'NDVR-ADM' area
ABADMPSZ = 5064         Page size for 'NDVR-ADM' area
ABADMNPG = 1000         Number of pages (blocks) in 'NDVR-ADM' area

ABLOGDSN = @DBPREFIX@.CCDB.CCDBLOG   Dsname for 'NDVR-LOG' file
ABLOGDD  = CCDBLOG      File name (DD name) for 'NDVR-ADM' file
ABLOGUNI = @UNIT@       Physical unit type (e.g. 3390)
ABLOGSER = @PACK@       Physical disk volume for 'NDVR-LOG' file
ABLOGLOW = 905001       Starting page for 'NDVR-LOG' area
ABLOGPSZ = 5064         Page size for 'NDVR-LOG' area
ABLOGNPG = 1000         Number of pages (blocks) in 'NDVR-LOG' area

ABPAKDSN = @DBPREFIX@.CCDB.CCDBPAK   Dsname for 'NDVR-PAK' file
ABPAKDD  = CCDBPAK      File name (DD name) for 'NDVR-PAK' file
ABPAKUNI = @UNIT@       Physical unit type (e.g. 3390)
ABPAKSER = @PACK@       Physical disk volume for 'NDVR-PAK' file
ABPAKLOW = 909001       Starting page for 'NDVR-PAK' area
ABPAKPSZ = 5064         Page size for 'NDVR-PAK' area
ABPAKNPG = 3            Number of pages (blocks) in 'NDVR-PAK' area
```

* GJTPINTC = " Specify the TP system name to be used for IDMSINTC * If left null, the CICS System ID is used internally. * GJTPFET = CICS Specify the TP system name to be used for the * front end table. This cannot be null. *

Appendix D. SMP/E

D.1 Overview of SMP/E D-4
D.1.1 Non-SMP/E Install D-4
D.1.2 Target and Distribution Libraries D-4
D.1.3 SMP/E Invokes Utilities D-5
D.1.4 MCS D-5
D.1.5 CSI D-5
D.1.6 DDDEFs D-7
D.1.7 Indirect Libraries D-7
D.1.8 JCL D-7
D.1.9 RECEIVE/APPLY/ACCEPT D-8
D.1.10 FMID D-9
D.1.11 SYSMOD D-10
D.1.12 Sample SMP/E job D-10
D.1.13 CSI Entries D-11
D.1.14 CHECK Parameter D-12
D.1.15 Reversing Processing D-12
D.1.16 LMOD D-12
D.1.17 SOURCEID D-13
D.1.18 Relfiles D-14
D.1.19 LIST Command D-14
D.1.20 Restarting an SMP/E Job D-15
D.1.21 In Conclusion D-15
D.2 Debugging SMP/E Jobs D-16
D.2.1 Order of Processing D-16
D.2.2 Output Datasets D-16
D.2.3 Error Message Flag D-17
D.2.4 Syntax/Conflict Checking D-17
D.2.5 Copy Failure D-17
D.2.6 Zap Failure D-17
D.2.7 Assembly Failure D-18
D.2.8 Link Failure D-18
D.2.9 SMPLOG D-18
D.2.10 Recovery D-18
D.2.11 Support D-19
D.2.12 ASMA417C D-19
D.2.13 SYSLIB D-19
D.2.14 MODID Error D-20
D.2.15 MODID Warning D-21
D.2.16 Superceding (SUP) APARs and Test Fixes D-22
D.2.17 Removing APARs D-22

This appendix provides introductory information regarding SMP/E. It begins with an overview of SMP/E, explaining the terminology and concepts, the SMP/E components, and sample SMP/E jobs. It concludes with information for debugging SMP/E jobs, sources of information and how to resolve errors.

D.1 Overview of SMP/E

This section presents an introduction to SMP/E terminology and concepts. It is intended for those who are responsible for installing and/or maintaining CA-IDMS but are unfamiliar with SMP/E. Its purpose is to introduce the SMP/E commands used in the maintenance and install jobs provided by Computer Associates.

D.1.1 Non-SMP/E Install

SMP/E is an IBM utility to install software into libraries. It is not involved in the database portion of the CA-IDMS install.

The easiest way to introduce SMP/E is to begin by comparing an SMP/E install to a non-SMP/E install. Consider a simple non-SMP/E install. You have a tape of modules (macros, source and object). You allocate some libraries. You run some utilities (IEBCOPY, assemblies, links) to process the modules and populate the libraries. When you are done, you have a loadlib that you can put in your startup deck.

D.1.2 Target and Distribution Libraries

The first difference between a non-SMP/E install and an SMP/E install is that SMP/E uses two sets of libraries: the target libraries and the distribution libraries. The target libraries correspond to the libraries that you allocate in a non-SMP/E install. You put the target loadlib in your startup deck. The target libraries are the executable libraries. If you make a bad modification to the target libraries, users notice.

The distribution libraries are the backup libraries. Every element (source, macro, object) that SMP/E installs has a copy in a distribution library.

The term element can mean a source module, a macro or an object module. The distribution libraries contain only elements. Executable load module are not elements, but they are created by combining elements.

The distribution loadlib is not an executable loadlib. It is comprised of object modules that are individually link-edited. In contrast, the target loadlib is created by linking object modules together to form executable load modules. The target loadlib is like your car; that is, you can run it. The distribution loadlib is like a repository of spare parts.

So the first difference between an SMP/E install and a non-SMP/E install is that with SMP/E, two sets of libraries are allocated rather than one. The distribution libraries are not copies of the target libraries and the distribution loadlib cannot substitute for the target loadlib. Job 1 of the install allocates the target and distribution libraries, as well as some additional SMP/E files.

SMP/E uses the distribution libraries to remove changes from the target libraries. Suppose you use SMP/E to install a zap for IDMSDBIO but then you decide to

remove the zap. The zap has updated your target loadlib but not your distribution loadlib. You tell SMP/E to take off the zap. SMP/E knows that the zap hits IDMSDBIO, so it takes IDMSDBIO from the distribution loadlib and links it into the target loadlib, thereby erasing the zap. With SMP/E, updates are removed by linking or copying the affected module(s) from the backup (distribution) library into the executable (target) library. Note that zap modifications are removed, not by reverse zaps, but by link edit module replacements. This means if a module has been updated by two zaps, SMP/E can remove both of them or neither of them, but not just one of them.

D.1.3 SMP/E Invokes Utilities

The second difference is that you do not run various utilities to populate these libraries with modules. You run one utility, PGM=GIMSMP, which is the SMP/E program. SMP/E can transfer control to other utilities; for example, IEBCOPY. During the SMP/E CA-IDMS install, your libraries are still populated with modules created by copies, assemblies, and links. But the only EXEC card in your JCL to accomplish this processing is the EXEC card that identifies PGM=GIMSMP.

D.1.4 MCS

How does SMP/E know about the modules on the tape and the processing required to install them? The person who puts the modules on the tape also has to supply a recipe or blueprint that identifies all the modules and describes how they are to be installed. This file is called the modification control statements (MCS) and is always the starting point of SMP/E installation or maintenance. The MCS is a sequential file. By Computer Associates' convention, the MCS file is always File 32 on the install tape and is always named SMPMCS. The MCS describes each element:

- The module's type (macro, source, object)
- The module's location on tape or disk file
- The utilities needed to install the module
- The libraries where it should be installed

With SMP/E, every installation and every update made to your system (for example, APARs and PTFs) is be defined by MCS. Products that are not designed for installation with SMP/E cannot be installed using SMP/E because there is no MCS. The MCS is the director of the show.

D.1.5 CSI

An advantage to using SMP/E is that it preserves a record of all installation and maintenance processing it performs. SMP/E uses a big VSAM dataset called the Consolidated Software Inventory (CSI) and seven additional SMP/E datasets (SMPPTS, SMPLTS, SMPMTS, SMPSTS, SMPSCDS, SMPLOG, SMPLOGA) to maintain this information. The preservation of information in these additional datasets is another difference between an SMP/E install and a non-SMP/E install.

The CSI contains information about every module, every link, every library, and every utility as well as information about every process which has been accomplished and whether the process was successful. When you ask SMP/E to do something, it first checks your MCS against the information in the CSI to ensure you are not doing something that conflicts with an earlier change. For instance, SMP/E prevents you from installing one version of a module if a later version is already installed. If a process to be initiated by the MCS is not consistent with information in the CSI, SMP/E won't do it. If the MCS is the director of the show, the CSI is the authority that says whether the show can go on.

The CSI is divided into zones. The three types of zones:

- **Global Zones** contain information not specifically related to the target or distribution libraries; for example, the names of the utilities.
- **Target Zones** contain information about the target libraries and the modules in the target libraries.
- **Distribution Zones** contain information about the distribution libraries and the modules in the distribution libraries.

The CSI can have only one global zone and its name must be GLOBAL. You can think of the global zone as the entryway into the CSI, containing the names and locations of the other zones.

The CA-IDMS install creates one CSI with three zones: a global zone, a target zone, and a distribution zone. Their names are GLOBAL, IDMSTGT and IDMSDLB. CA-IDMS Tools and CA-Endevor/DB must be installed into the same zones. Other Computer Associates' products may use zone names of GLOBAL, CAITGT and CAIDLIB.

Each installed element has an entry in the target zone (IDMSTGT) and an entry in the distribution zone (IDMSDLB). Object modules have MOD entries, source modules that are assembled have SRC entries, and all other elements have MAC entries. Each MOD, SRC, and MAC entry in the CSI contains information about a particular element.

Example: The MOD entry for ADSOPLFE might show:

- the libraries containing ADSOPLFE
- if a new version has been installed by a maintenance tape
- if it has had zap updates
- the product which owns ADSOPLFE
- the load modules that include ADSOPLFE

D.1.6 DDDEFs

The CSI and SMPxxx datasets are allocated in Job 1 of the CA-IDMS install. Job 1 also initializes the CSI with the GLOBAL, IDMSTGT and IDMSDLB zones, saves the names of the utilities in the GLOBAL zone, and creates DDDEFs. A DDDEF is like a dynamic DD statement because it connects a ddname to a dataset name. The CA-IDMS, CA-IDMS Tools and CA-Endevor/DB installs create DDDEFs in the CSI for every target, distribution and SMPxxx dataset. Installation variables allow you to change the dataset names, but not the ddnames.

D.1.7 Indirect Libraries

The install tape contains the modules for all the installable products. In Job 2 of the install, CAIIPDS downloads the modules for the products you are installing into two libraries (a source library and an object library) called the indirect srclib and the indirect objlib. These indirect libraries are neither target libraries nor distribution libraries. They are needed for the install and can be deleted after the install is completed. The modules are downloaded into the indirect libraries before SMP/E is invoked to run the utilities that populate the target and distribution libraries. Instead of reading the modules from the tape, SMP/E reads CA-IDMS modules from the indirect files. The indirect files are input files for SMP/E. SMP/E cannot read the modules directly from the tape because the modules on the base tape are encrypted and compressed.

No DDDEFs are created for the indirect srclib and the indirect objlib because these libraries are used only for the base install and should be deleted after it has been completed. They are not used after the ACCEPT job has been run.

Note: Since maintenance tapes do not update the indirect files, they contain obsolete modules once maintenance is installed.

D.1.8 JCL

Every SMP/E job needs an SMPCSI DD statement in the JCL to point to the CSI containing the GLOBAL zone. SMP/E reads the CSI to get the DDDEFs to point to the target and distribution libraries. Because of the DDDEFs, the JCL to run SMP/E for CA-IDMS is relatively short.

What does the JCL to run an SMP/E install look like? We need an EXEC card for PGM=GIMSMP. The CSI is pointed to by the SMPCSI DD statement. The MCS is pointed to by the SMPPTFIN DD statement. We need an INDSRC DD statement to point to the indirect srclib and an INDOBJ DD statement to point to the indirect objlib. The DDDEFs point to all the other datasets.

So now we know what program to run on the EXEC statement. We also know how to point to the MCS, the CSI, the empty datasets, and the elements to fill them. We need a command that says, "Okay, I'm pointing to everything you need, so read the MCS and do the install." We need something similar to an INSTALL command.

D.1.9 RECEIVE/APPLY/ACCEPT

Most utilities use a specific ddname to point to the dataset containing the commands for a particular invocation. For example, the IEBCOPY utility uses a SYSIN dataset for its COPY command(s) and the linkage editor uses a SYSLIN dataset for its various commands.

You know that SMP/E uses the SMPCSI ddname to point to the CSI, and the SMPPTFIN ddname to point to the MCS. You might assume there is a third ddname to point to something like an INSTALL command. Does SMP/E have something like an INSTALL command in a SYSIN dataset? Yes, almost. Instead of a SYSIN dataset, SMP/E uses an SMPCNTL dataset for commands. And instead of an INSTALL command, we have the RECEIVE, APPLY and ACCEPT commands. After you have run the RECEIVE and APPLY and ACCEPT commands one after the other, you have completed the install process. These three commands use the MCS in different ways and are usually run in separate jobs, but their combined effect is like an INSTALL command.

Each install runs a RECEIVE job, an APPLY job, and an ACCEPT job. When these jobs have completed, the SMP/E portion of the install is virtually complete. The remaining install jobs deal with the CA-IDMS database or minor modifications to the SMP/E environment.

It is not a coincidence that there are three SMP/E install commands and three zones in the CSI.

- The RECEIVE command:
  - Checks the MCS for syntax errors
  - Copies the MCS into the SMPPTS dataset
  - Updates the GLOBAL zone
- The APPLY command:
  - Reads the MCS from the SMPPTS library
  - Creates MOD, SRC, MAC and LMOD entries in the target zone
  - Runs utilities to populate the target libraries from the elements in the indirect files.
- The ACCEPT command:
  - Reads the MCS from the SMPPTS
  - Creates MOD, SRC, and MAC entries in the distribution zone
  - Runs utilities to populate the distribution libraries from the elements in the indirect files
  - Deletes the MCS from the SMPPTS library

Each of these three commands must be preceded by a statement that sets the boundary to the right zone for the command.

SMP/E Command	Type of Zone	CA-IDMS Zone Name (Boundary)
RECEIVE	Global	GLOBAL
APPLY	Target	IDMSTGT
ACCEPT	Distribution	IDMSDLB

D.1.10 FMID

A base install tape contains all the modules for the CA-IDMS product line. There is only one MCS file and it describes all the products' modules, yet products can be installed individually. This is because the MCS file on the base tape can be thought of as a series of smaller, logical files beginning with ++FUNCTION statements in the same way that input to the zap utility can be thought of as a series of individual zaps beginning with NAME statements. The units in the MCS beginning with ++FUNCTION statements are called function SYSMODs. Each function SYSMOD typically describes the modules belonging to one CA-IDMS product. For example, to install CA-ADS, you would RECEIVE, APPLY, and ACCEPT the function SYSMOD corresponding to CA-ADS.

The name of a function SYSMOD is called it's FMID. The FMID of the CA-ADS SYSMOD is CFEF000. When you tell SMP/E to RECEIVE CFEF000, SMP/E creates a member in the SMPPTS named CFEF000 and copies that section of the MCS into it. When you tell SMP/E to APPLY CFEF000, SMP/E installs the modules that belong to the CFEF000 section of the MCS into the target libraries.

The terms *function*, *function SYSMOD* and *FMID* can be used somewhat interchangeably, although the 7-character name of a function SYSMOD is always called its FMID.

The FMID naming convention used for CA-IDMS:

1. The first character is always C.
2. The second and third characters are the two-character product code.
3. The fourth and fifth characters are the release number (F0 for Release 15.0).
4. The sixth and seventh characters are usually 00, but other numbers are used when a product requires more than one function SYSMOD.

D.1.11 SYSMOD

Function SYSMODs are found only on the base tape and are used to install modules for the first time. There are three other types of SYSMODs: APARs, PTFs, and USERMODs. These three types of SYSMODs typically make modifications to modules which were previously installed by function SYSMODs. An APAR always begins with a ++APAR statement, a PTF always begins with a ++PTF statement and a USERMOD always begins with a ++USERMOD statement. A SYSMOD always begins with one of these four statements, and the statement identifies the 7-character name of the SYSMOD. Each SYSMOD name must be unique.

A PTF, APAR or USERMOD must specify one FMID in its MCS and any elements it affects cannot belong to another FMID. If you have not installed a particular function SYSMOD, SMP/E does not let you install an APAR, PTF, or USERMOD which specifies its FMID. In other words, you cannot update a product you have not installed.

D.1.12 Sample SMP/E job

This example depicts how SMP/E works. It shows how three products could be installed with SMP/E. Each SMP/E command ends with a period and may span more than one line. The amount of spacing between parameters does not matter, and parameter values can be separated by blanks, commas or both. Comments begin with **/*** and end with ***/**. Comments must be included within the body of a command; they cannot be inserted between commands.

Note: This example is for illustrative purposes only and would not run successfully because some of the SYSMODs shown require other SYSMODs which are not shown. It also shows the three SMP/E commands being run in one job step, which is not good practice. The RECEIVE, APPLY and ACCEPT commands should be run in separate jobs, or at least separate steps with conditional checking.

```
//INSTALL  EXEC PGM=GIMSMP,REGION=4096K,PARM='DATE=U'
//SMPCSI   DD DISP=SHR,DSN=IDMS.R150.CSI
//SMPPTFIN DD DSN=SMPMCS,UNIT=TAPE,LABEL=32,
//            VOL=SER=F0GJ0B,DISP=OLD
//INDSRC   DD DISP=SHR,DSN=IDMS.R150.INDSRC
//INDOBJ   DD DISP=SHR,DSN=IDMS.R150.INDOBJ
//SMPCNTL  DD *
     SET BOUNDARY ( GLOBAL ).
     RECEIVE SELECT(
                     CGJF000   /* CA-IDMS/DB */
                     CGQF000   /* CA-IDMS/DC */
                     CFEF000   /* CA-ADS    */
                   )
                   .
     SET BOUNDARY (IDMSTGT) .
     APPLY SELECT  ( CGJF000 CGQF000 CFEF000 ).
     SET BOUNDARY(IDMSDLB).
     ACCEPT SELECT(CGJF000,CGQF000,CFEF000).
```

1. The first command is a *SET BOUNDARY* command. SMP/E opens the CSI dataset named on the SMPCSI DD statement in the JCL and accesses its GLOBAL zone.
2. The next command is a *RECEIVE* command specifying three SYSMODs. SMP/E opens the dataset named on the SMPPTFIN DD statement. It expects to find a sequential dataset containing concatenated SYSMODs, including CGJF000, CGQF000, and CFEF000. They could be function SYSMODs, PTFs, APARs or USERMODs. SMP/E checks the SYSMODs for syntax errors, then copies each of them into a separate member in the SMPPTS library. The member name is the SYSMOD name.
3. The next command is another *SET BOUNDARY* command, this time for IDMSTGT. SMP/E accesses the GLOBAL zone to find the location of the IDMSTGT zone and accesses the IDMSTGT zone.
4. The *APPLY* command specifying CGJF000, CGQF000 and CFEF000 causes SMP/E to read the three members in the SMPPTS library to obtain the install instructions.
5. Following these instructions (the MCS), SMP/E updates the IDMSTGT zone and runs utilities to populate the target libraries with the elements named in the MCS.

D.1.13 CSI Entries

Each element (source, object or macro) described in the MCS causes SMP/E to create a SRC, MOD or MAC entry in the CSI to describe the element.

Example: These are examples of element entries: a MAC entry for #DCPARM, a MOD entry for CULLPOPT, and a SRC entry for RHDCPARM.

```
#DCPARM   LASTUPD         = UCLIN    TYPE=UPD
          LIBRARIES       = DISTLIB=DISTMAC
          FMID            = CGJF000
          RMID            = CGJF000
          GENASM          = RHDCPARM

CULLPOPT  LASTUPD         = CFKF000  TYPE=ADD
          LIBRARIES       = DISTLIB=DISTLOAD
          FMID            = CFKF000
          RMID            = FK00SP2

RHDCPARM  LASTUPD         = CGJF000  TYPE=ADD
          LIBRARIES       = DISTLIB=DISTSRC
          FMID            = CGJF000
          RMID            = CGJF000
```

D.1.14 CHECK Parameter

A CHECK parameter can be added to the APPLY, ACCEPT and RESTORE commands. This makes the commands read only; no updates are made to the libraries. SMP/E reads the MCS and checks it against the CSI to see if there are any inconsistencies between what you are trying to do and what has already been done.

Note: When updating your system, run an APPLY CHECK command before you run the APPLY command. If you are going to run an ACCEPT command, you should run an ACCEPT CHECK first.

D.1.15 Reversing Processing

A RESTORE command reverses the effects of an APPLY command. If you APPLY an APAR that is bad, you can remove it from your system with the RESTORE command. The RESTORE reverses an APPLY by linking the affected modules into the target library from the distribution library. The RESTORE command modifies the target libraries and the boundary for the RESTORE command is IDMSTGT. A CHECK parameter is available for the RESTORE command; always run a RESTORE CHECK before running a RESTORE.

Warning: A SYSMOD cannot be RESTOREd after it has been ACCEPTed because the ACCEPT updates the distribution libraries.

The RECEIVE command can be reversed with a REJECT. You do not often need to run a REJECT because running a RESTORE automatically invokes a REJECT unless you modify your CSI to turn on the NOREJECT option. The CA-IDMS install does not turn on this option.

Zone Type	Boundary Name	Command	Reverse Command	CHECK Parameter Available
Global	GLOBAL	RECEIVE	REJECT	NO
Target	IDMSTGT	APPLY	RESTORE	YES
Distribution	IDMSDLB	ACCEPT	(none)	YES

D.1.16 LMOD

For every object module that SMP/E installs, a MOD entry in the CSI is created. If the object module is included in a link, SMP/E creates an LMOD subentry under the MOD entry to show the name of the load module. An LMOD subentry may contain more than one name if the object module participates in more than one load module, but for each load module, the only information shown in the MOD entry is its name.

```
RHDCHPCS  LASTUPD         = CGJF000  TYPE=UPD
          LIBRARIES       = DISTLIB=DISTLOAD
          FMID            = CGJF000
          RMID            = GJ00SP2
          UMID            = LO78047   LO78048   LO81586
          LMOD            = IDMSDC    IDMSDCT   IDMSDCTU  IDMSDCU
```

For each load module that SMP/E creates, an LMOD entry is created in the CSI. This should not be confused with the LMOD *subentry* created under the MOD entry. An LMOD subentry is part of the MOD entry and shows only one or more load module names. An LMOD entry is created for a load module and shows the load module name, ORDER statements, MODE statements, ENTRY statements, whether the link is reentrant, and the library that contains the load module.

Example.: This example shows an LMOD entry for IDMSSERV.

```
IDMSSERV  LASTUPD         = CGJF000  TYPE=ADD
          SYSTEM LIBRARY  = LOADLIB
          LKED ATTRIBUTES = RENT,NCAL
          LKED CONTROL    =  ORDER   IDMSSERV
                             ENTRY   #EPMAP
                             MODE    AMODE(31),RMODE(ANY)
```

D.1.17 SOURCEID

The example is another version of the RECEIVE command. It is similar to the RECEIVE command used for maintenance tapes.

Example:

```
SET BOUNDARY(GLOBAL).
RECEIVE.
```

There is no SELECT operand on this RECEIVE command. No SYSMODs are specifically selected to be RECEIVEd. In this case, SMP/E:

1. Reads your CSI to determine which FMIDs you have installed
2. Scans the MCS pointed to by the SMPPTFIN DD statement looking for PTFs
3. RECEIVEs each PTF that specifies one of those FMIDs on its ++VER statement

In other words, a RECEIVE command with no operands RECEIVEs all PTFs in the MCS that fit your system.

There is a SOURCEID parameter that can be used on the RECEIVE command. The SOURCEID parameter specifies a name, and in the above example, SMP/E would assign this name to each PTF RECEIVEd. It creates an entry in the CSI for the SOURCEID name and that entry shows all the PTFs that belong to this SOURCEID. If you looked at the SOURCEID entry in the CSI, you would see the names of the PTFs that were RECEIVEd.

The RECEIVE job that is delivered in SAMPJCL on CA-IDMS maintenance tapes uses the SOURCEID parameter with a name that shows the genlevel. Any maintenance that fits your system is RECEIVEd.

```
SET BOUNDARY(GLOBAL).
RECEIVE SOURCEID(GJ0011).
```

The APPLY and ACCEPT jobs on CA-IDMS maintenance tapes also use the SOURCEID parameter instead of the SELECT parameter and the SOURCEID parameter again specifies the name that was used on the RECEIVE command. However, the SOURCEID parameter means something different here. On the APPLY and ACCEPT statements, the SOURCEID parameter tells SMP/E to install the SYSMODs that have already been assigned to this name.

Suppose the RECEIVE job specifies a SOURCEID of FROG and RECEIVEs four SYSMODs: A, B, C, and D. An APPLY job with no SELECT parameter but with a SOURCEID parameter for FROG would do the same thing as an APPLY job with a SELECT parameter specifying A, B, C and D.

D.1.18 Relfiles

Indirect files are disk libraries used for input during the APPLY and the ACCEPT. Indirect files are used on base tapes, but maintenance tapes use relfile processing. When relfile processing is used, the RECEIVE job allocates datasets called SMPTLIB datasets and downloads modules from the tape into them. The APPLY and ACCEPT jobs use the modules in the SMPTLIB datasets in the same way that they use modules from the indirect files. The ACCEPT job, if successful, deletes the SMPTLIB datasets.

The main difference between relfile processing and indirect file processing is who allocates libraries for downloaded modules and who copies the modules from the tape into those libraries. With relfile processing, this is all done by SMP/E during RECEIVE processing. With indirect file processing, it is done outside of SMP/E before the SMP/E jobs are executed. Base tapes for CA-IDMS, CA-IDMS Tools and CA-Endevor/DB use indirect file processing because the modules on the tape are in a format that SMP/E cannot read. The MCS tells SMP/E whether indirect files, relfiles, or both are to be used.

D.1.19 LIST Command

A very helpful SMP/E command is the LIST command. You can use the LIST command to look at information in the CSI. The boundary that you use depends on which zone contains the information you want to see. (If you are having trouble with an APPLY or RESTORE job, look in the target zone.)

Example: Examples of the LIST command:

```
LIST MAC(#PMOPT).
LIST SYSMOD(CGJF000).
LIST SYSMOD ERROR.
LIST DDDEF(SYSLIB).
LIST MOD(IDMSDBIO).
LIST LMOD(IDMSDC).
```

Example: To determine the "links" for RHDCCTAB:

```
SET BOUNDARY (IDMSTGT).
LIST LMOD(RHDCCTAB) XREF.
```

D.1.20 Restarting an SMP/E Job

If you get an error in an SMP/E job, D.2, “Debugging SMP/E Jobs” on page D-16 can help you resolve it. Then you must know how to continue processing.

If you are running a job with the CHECK parameter, just submit the job again.

If the job that failed has a SOURCEID parameter and doesn't have a SELECT parameter, you can just resubmit the job. SMP/E will not attempt to reprocess SYSMODs that were successful.

If you are running a RECEIVE, APPLY, ACCEPT, RESTORE, or REJECT job with a SELECT parameter, you may need to modify the command before you resubmit the job. Check the STATUS section in the output of the job that failed. If any SYSMODs were successful, remove them from the SELECT list before resubmitting the job. Do not use REDO.

D.1.21 In Conclusion

To keep your systems operating smoothly, these policies should be incorporated into your environment:

1. Always have a good system backup
2. Never use BYPASS(ID) on a RESTORE command
3. Do not ACCEPT USERMODs
4. Do not ACCEPT optional APARS

If you possess an extensive knowledge of SMP/E processing, these guidelines may be ignored in limited circumstances:

- Do not use REDO
- Do not use BYPASS(ID)
- Do not use UCLIN (except for DDDEFs, OPTIONs, and GENASM)
- Do not ACCEPT test fixes or APARs

D.2 Debugging SMP/E Jobs

This section discusses debugging techniques that may be useful in determining the problem when an SMP/E job with an enormous amount of output fails. It also supplies some information about how to recover from an error after you have determined what the error is.

D.2.1 Order of Processing

A typical SMP/E job starts by checking your MCS against the CSI. If your MCS seems reasonable, SMP/E proceeds to do some or all of the following:

- Update the CSI (JCLIN processing)
- Print messages (JCLIN processing successful)
- Call utilities (superzap, IEBCOPY, assembler, linkage editor)
- Query the results from the utilities (condition code checking)
- Print more messages (for example, assembly processing succeeded or failed)
- Make final updates to the CSI (changing RMID, adding LMOD, saving status of SYSMOD)

SMP/E always does processes in the following order:

1. MODID checking
2. JCLIN processing
3. IEBCOPY
4. Assemblies
5. Links

If there are any errors in one of these steps, the remaining processing is not done. For example, if there is a JCLIN error, no utilities are called and there is no SYSPRINT dataset.

D.2.2 Output Datasets

In the output listing of your job, the SMP/E messages are printed to the SMPOUT, SMPRPT, and/or SMPLIST datasets. The output from the utilities (zaps, copies, assemblies, links) are printed to output datasets such as SYSPRINT. If you are using SDSF, you can separate the output datasets by selecting your output job with a **?** instead of an **S**.

D.2.3 Error Message Flag

SMP/E prints many messages about successful processing, but it flags error messages with double asterisks (**). To determine what type of error occurred, find the first double asterisk in the output.

D.2.4 Syntax/Conflict Checking

An error can occur when SMP/E is checking the MCS against the CSI, or later, when a utility is running. If the first SMP/E error message (GIMxxxxxE) says nothing about copy processing, zap processing, assembler processing, link processing, or update processing, the error did not occur in a utility and the SMP/E messages directly after the first error message give more information about the error that occurred. If the job that failed is a RECEIVE job, the FMID may be wrong (see D.1, “Overview of SMP/E” on page D-4) or a prerequisite may be missing (see D.2.14, “MODID Error” on page D-20). If it is an APPLY job, see D.2.14, “MODID Error” on page D-20. If it is a RESTORE job, see D.2.17, “Removing APARs” on page D-22.

D.2.5 Copy Failure

If the GIMxxxxxE message says that copy processing failed, go to the SYSPRINT dataset and find the IEBCOPY message for the module mentioned in the GIMxxxxxE message. Perhaps the module doesn't exist in the library that IEBCOPY was using as input. Perhaps the directory of an output library is too small. Perhaps there was a B37 abend.

D.2.6 Zap Failure

If the GIMxxxxxE message says that zap processing failed, go to the SYSPRINT dataset to find the message from the zap utility. If the verify failed, see the information in the D.2.16, “Superceding (SUP) APARs and Test Fixes” on page D-22.

If the AMAxx message is a warning, there are two common causes:

1. IDRDATA may be full.
2. There may be a blank line after the last REP statement.

A warning in the zap utility causes an error in SMP/E.

Resolution.

1. RESTORE your APAR or test fix
2. Either:
   - Relink the load module to eliminate the IDRDATA message Or
   - Remove the blank line from the bottom of your APAR or test fix
3. Reinstall the APAR or test fix

D.2.7 Assembly Failure

If the GIMxxxxxE message says assembly processing failed, search the SYSPRINT dataset for **ASMA**. If you do not find an error, search for **STATEMENTS FLAGGED** and work backwards.

If the assembly error is:

- ASMA417C — see D.2.12, “ASMA417C” on page D-19
- An unidentified op code — the problem may be the SYSLIB concatenation. See D.2.13, “SYSLIB” on page D-19.

Otherwise, the problem may be a macro that requires CICS preprocessing.

D.2.8 Link Failure

A GIM23901E message states link processing failed for *MODULE xxxxxxxx IN LMOD xxxxxxxx*. Note the name of the load module (LMOD) in the message; in the SYSPRINT or SMPOOxxx portion of the output listing, search for *load-module-name(R)*.

Example: If the message is:

```
GIM23901v LINK-EDIT PROCESSING FOR SYSMOD CGJF000 FAILED
                     FOR MODULE IDMSBRBK  IN LMOD IDMSWARM
```

search for IDMSWARM(R) and scroll up a few lines to find the error.

Alternatively, you could search for known common errors such as IEW2456E, IEW2303S and IEW2648E.

D.2.9 SMPLOG

SMP/E records processing information in the SMPLOG. As long as the SMPLOG DDDEF or DD statement points to the SMPLOG dataset, SMP/E writes information to the end of the log when an SMP/E job modifies your system. When the SMPLOG fills up, SMP/E writes to SMPLOGA, the alternate log. Generally, the information in the CSI is more helpful than the information in the log; you can browse the log. If you turn hex on, you can see the date and time on the left.

D.2.10 Recovery

After you have fixed a problem, try installing the SYSMOD again. You do not need REDO when you APPLY or ACCEPT a SYSMOD that has previously failed. If the problem was caused by the environment and there was no problem with the SYSMOD itself, just APPLY or ACCEPT it again. If the SYSMOD contains an error:

1. RESTORE the SYSMOD
2. Correct it

3. Reinstall it

D.2.11 Support

If you have tried these suggestions and are unable to resolve your problem, please have the following information available when you call the support line:

- The current genlevel of your target libraries
- The current genlevel of your distribution libraries
- The name of the failing SYSMOD
- The type of SYSMOD (function, APAR, PTF, USERMOD)
- The complete text of the first error message from SYSPRINT
- For copies and assemblies, the name of the module
- For links, all the link control cards for the first failing link
- For non-utility errors, the first GIMxxxxxE message and the text of the information messages following it

D.2.12 ASMA417C

When your site changes from assembler IEV90 to ASMA90, you must update the assembler parms in the CSI.

```
SET BOUNDARY(GLOBAL).
UCLIN.
REP UTILITY(IDMSASM) PARM(NOXREF,NOOBJ,DECK,NORLD).
ENDUCL.
```

If the assembly failed with an ASMA417C, it should run correctly after making this change.

D.2.13 SYSLIB

Only the macros stored in libraries concatenated in the SYSLIB DD statement are available to the assembler. To see which libraries are in your SYSLIB concatenation, look for SYSLIB in the File Allocation Report in your output.

The SYSLIB concatenation should contain these libraries, in this order:

1. SYS1MAC
2. SMPMTS
3. DISTMAC
4. CEEMAC
5. CICSMAC
6. CWIF0MLD

7. CABF0MLD
8. AMODGEN

If the concatenation is wrong, use this job as a guide for correcting it:

```
SET BOUNDARY(IDMSTGT)   /* or IDMSDLB */.
UCLIN.
REP DDDEF(SYSLIB)
   CONCAT( SYS1MAC
           SMPMTS
           DISTMAC
           CEEMAC
           CICSMAC
           CWIF0MLD
           CABF0MLD
           AMODGEN
          ).
ENDUCL.
```

Use the target zone if an APPLY job is failing. Use the distribution zone if an ACCEPT job is failing. Ultimately, the target and distribution zones should have the same SYSLIB concatenation.

D.2.14 MODID Error

When you install maintenance on your system, often one or more object modules are replaced. You don't want to replace any module with an obsolete version. You don't want to back-level any of your software. SMP/E is designed such that prevention of back-leveling is a high priority.

SMP/E can't use SYSMOD names to distinguish which one of two SYSMODs has a newer version of a module. However, if one SYSMOD contains a reference to the other SYSMOD in its MCS, it must have been written later and therefore must be delivering a newer version of the module. This is the logic behind the rule that once SYSMOD A replaces a module, you can't install SYSMOD B that replaces the same module unless SYSMOD B references SYSMOD A. SYSMOD B needs to reference SYSMOD A so that SMP/E knows that SYSMOD B is delivering a newer module. If SYSMOD B doesn't reference SYSMOD A, it might replace the module with an older version.

Suppose SYSMODA is a PTF installed on your system and it installed a replacement for module IDMSDBIO. When SMP/E installed SYSMODA, it updated the MOD entry for IDMSDBIO in the CSI to show an RMID subentry containing the name of SYSMODA. Now the MOD entry for IDMSDBIO contains both an FMID subentry specifying the name of the function SYSMOD that originally installed IDMSDBIO, and an RMID subentry showing the name of the SYSMOD that last replaced IDMSDBIO, in this case SYSMODA.

```
MOD IDMSDBIO
    FMID CGJF000
    RMID SYSMODA
```

Any future SYSMODs that replace IDMSDBIO must reference IDMSDBIO's RMID. The reference must be either a PRE or SUP parameter on the ++VER statement in the MCS. If SYSMODB has a PRE for SYSMODA (specifies SYSMODA as a prerequisite), then SMP/E lets you install SYSMODB and IDMSDBIO's RMID changes from SYSMODA to SYSMODB. Future replacement SYSMODs would require a PRE or SUP for SYSMODB.

If you try to install a SYSMOD that updates or replaces a module and doesn't reference that module's RMID, SMP/E issues a MODID error and doesn't install the SYSMOD.

You can use the LIST command to look at a module's FMID (installing SYSMOD), RMID (replacing SYSMOD) and UMID(s) (updating SYSMODs).

```
SET BOUNDARY(IDMSTGT).
LIST MOD(IDMSDBIO).


IDMSDBIO  LASTUPD         = CGJF000  TYPE=ADD
          LIBRARIES       = DISTLIB=DISTLOAD
          FMID            = CGJF000
          RMID            = GJ00SP2
          LMOD            = IDMSDBIO
```

Each object module has a MOD entry in the CSI. Each MOD entry has one FMID subentry, one RMID subentry (which may be the same as the FMID if the module has never been on a maintenance tape), and zero or more UMID subentries showing the SYSMOD that updated it since it was last replaced.

When you attempt to install a PTF, APAR or USERMOD, SMP/E makes a list of every module your SYSMOD affects. It notes the RMIDs for each module on the list. If your PTF, APAR or USERMOD doesn't reference these RMIDs, SMP/E issues a MODID error and will not install it.

D.2.15 MODID Warning

SMP/E also notes any UMIDs for the affected modules. These UMIDs typically indicate zaps to a module. If you are installing a zap to a module and it doesn't reference other zaps on the module (UMIDs), SMP/E issues the same MODID message as before, including the same regression messages. However, the condition code will be 4, SMP/E installs the zap, and there is no regression. A MODID error with a warning condition code means only that a module is being zapped and the module has other, unreferenced, zaps already installed. You can ignore these messages. Do not modify the prerequisites to get a zero condition code.

D.2.16 Superceding (SUP) APARs and Test Fixes

A thorough discussion of test fixes and applying APARs is located in 12.1.1.3, “Test Fixes and Corresponding APARs” on page 12-5.

D.2.17 Removing APARs

SMP/E removes a SYSMOD by doing link edits. It notes which elements (source modules, macros and object modules) are affected by the SYSMOD by checking for ++SRC, ++MAC, ++MOD, ++ZAP, ++MACUPD and ++SRCUPD statements in the MCS of the SYSMOD. When CA-IDMS source or macros are affected by a RESTORE, SMP/E performs any necessary assemblies and links the resulting object modules into the target loadlib. When an object module is affected (the SYSMOD contained a ++MOD or ++ZAP statement), SMP/E links the module from the distribution loadlib into the target loadlib. Note that SMP/E removes zaps by doing link edits.

Suppose an object module has two zaps applied and you run a job to RESTORE only the second one. If SMP/E were to the link edit to remove the second zap, the first zap would also be eliminated. Therefore, in a case like this, your RESTORE CHECK job would fail. SMP/E cannot remove only one of the zaps.

There are two options:

1. Restore both zaps and reinstall the first one. (This is usually the best course.)
2. Accept the first zap if it is a published APAR and if it is not higher than the high APAR of the next maintenance tape you will install as discussed in section 12.3, “Installing Maintenance Tapes” on page 12-8.

In other words, if the next maintenance tape will not know about the zap, do not ACCEPT it. If the first zap meets these requirements and you ACCEPT it, it zaps the module in the distribution library. Then it will not be eliminated when you remove the second zap.

Suppose you have APPLYed a maintenance tape but not ACCEPTed it. Suppose you then APPLY a zap to a module that was replaced by the maintenance tape. For the same reasons discussed above, SMP/E cannot remove the zap until you ACCEPT the maintenance.

Appendix E. CICS Macros

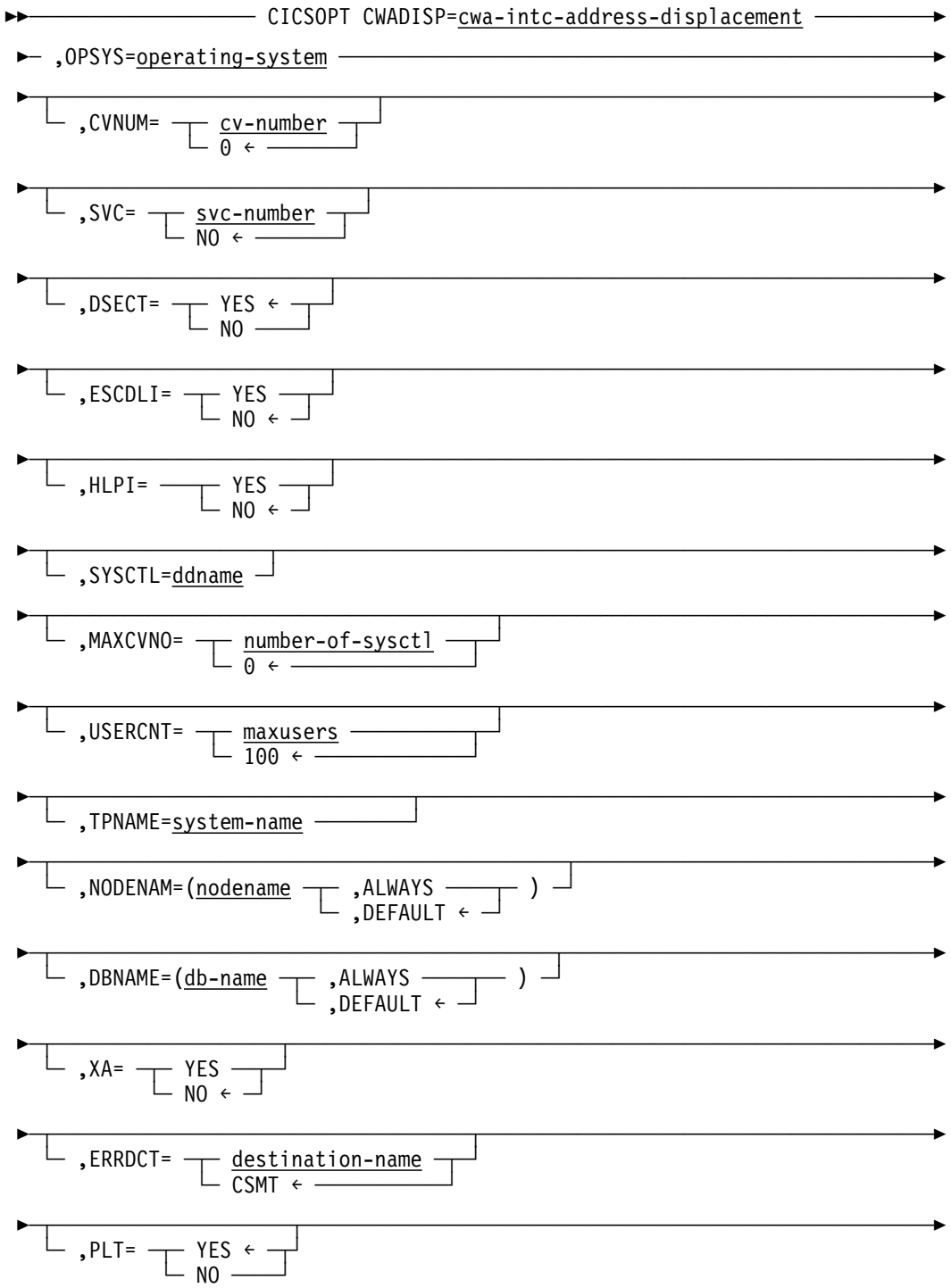
E.1 CICSOPT Macro E-4
E.2 IDMSCINT E-12

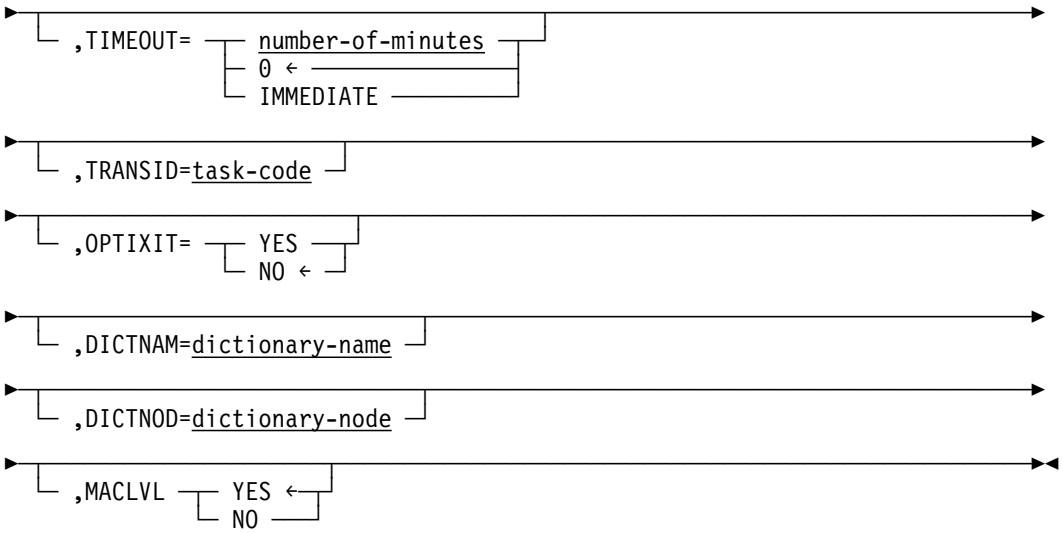
The CICSOPT macro is used to tailor the new CICS interface. This appendix describes the parameters that may be specified for the CICSOPT macro and instructions on its assembly and the creation of the IDMSINTC interface module.

►►For more information about the CICS macros, see the *CA-IDMS System Operations* manual.

E.1 CICSOPT Macro

CICSOPT syntax





CICSOPT Parameters:

CWADISP=

Identifies the displacement within the CICS CWA of a fullword to hold the address of the CICSOPT module. For cwa-intcddress-displacement, specify either a number of bytes (maximum value is 3584) or the name of a field within the CSA copy book.

Considerations: The specified field must be on a fullword boundary within the CWA and must be the same value given to the CWADISP parameter of the IDMSCINT macro.

OPSYS=

Identifies the operating system under which the DC/UCF system will run.

Valid values for operating-system:

- OS390
- VSE

CVNUM=

Identifies the number of the DC/UCF system to be accessed from CICS. For cv-number, specify the number used for the CVNUM parameter in the CV-gen.

SVC=

Identifies the number of the CA-IDMS SVC. For svc-number, specify a value as follows:

- If no SVC is being used, or if using SYSCTL, specify NO.
- If an SVC is being used by the DC/UCF system, specify the SVC number.

The SVC parameter is required if no SYSCTL file is specified.

DSECT=YES/NO

Specifies whether a DSECT only version of the CICSOPT macro is generated.

The parameter defaults to YES. However to generate a working CICS interface

module, the parameter DSECT=NO *must* be specified in the assembly of CICSOPT.

ESCDLI=

Is never explicitly specified. CDMSIJMP automatically sets this parameter to YES if installing CA-IDMS/DLI Transparency and the CICS interface (INT-CICS).

HLPI=

Specifies whether or not HLPI support is required for DL1.

SYSCTL=

Identifies the ddname of the file containing DC/UCF system control information.

If no SVC (described above) is specified, the SYSCTL parameter is required. Likewise, if SYSCTL is desired, the SVC parameter must be NO (SVC=NO).

MAXCVNO=

The number of extra SYSCTL DD cards are determined by the value specified for this operand. Valid values are in the range 0 through 9. The additional SYSCTL DDNAMES are derived by replacing either the eighth character of the SYSCTL ddname, or the leftmost blank character, with the numbers 1 through 9, as necessary.

USERCNT=

Specifies the maximum number of concurrent user sessions to CA-IDMS. It includes all active sessions, all suspended sessions, and all users that have not yet timed out. Valid values range from 1 to 100000.

TPNAME=

Specifies the name by which DC/UCF will identify all tasks running under this CICS system. For system-name, specify a four-character name.

This name forms the first part of the local transaction ID for database requests. It also forms the first four characters of the front end system ID for external request units. "BULK" is appended to system-name to form the front-end system ID. The front-end system ID is used for several things:

- Determines the packet size for communications
- Determines the maximum number of simultaneous requests from this CICS system to CA-IDMS.
- May also be used as an alternate task code for controlling external request unit processing.

If this parameter is omitted, it will default to the CICS sysid as defined during the CICS start-up.

NODENAM=

Identifies a system defined to the DC/UCF communications network to be contained in the CICSOPT module and the conditions under which programs signing on to the DC/UCF system will be directed to the named node for execution.

For nodename, specify the one- to eight-character name of a remote system. If the node name is not specified, the DC/UCF obtains the appropriate node name from the application program or from the SYSCTL file (OS/390 only).

ALWAYS
Indicates that nodename is to override any node named by the program. Requests from programs signing on to DC/UCF are always directed to the named node regardless of node name specifications made by the program.

DEFAULT
Indicates that requests from programs signing on to DC/UCF are to be directed to the named node only if the program does not name a node.

Note: Under OS/390 and VSE/ESA, SYSCTL node name specifications can override CICSOPT and program specifications.

DBNAME=
Identifies the database (or data dictionary) name to be contained in the CICSOPT module. This parameter also identifies the conditions under which programs signing on to the DC/UCF system access the named database.

For db-name, specify the name of the database that programs are to access when running under the DC/UCF system. If the database name is not specified, DC/UCF obtains the appropriate database name from the application program or from the SYSCTL file (OS/390 only).

ALWAYS
Indicates that db-name is to override any database named by the program. Programs signing on to DC/UCF always execute against the named database regardless of database name specifications made by the program.

DEFAULT
Indicates that programs signing on to DC/UCF are to execute against the named database only if the program does not name a database.

Note: Under OS/390 and VSE/ESA, SYSCTL database name specifications can override CICSOPT and program specifications.

XA=NO/YES
Designates whether the operating system is XA (YES) or not (NO). If you specify YES, CICSOPT allocates the primary user-oriented storage in the 31-bit storage area. This storage is retained across all successful task terminations for terminal-associated tasks, and this storage is reused on the next DC/UCF request. The storage is freed for any failing or non-terminal task.

MACLVL=YES/NO
Indicates whether applications using the CICS macro-level interface are supported by this CA-IDMS interface.

CAUTION:
If a macro-level application attempts communications with a CA-IDMS interface assembled with MACLVL=NO, the results are unpredictable.

ERRDCT=

Identifies the CICS transient data destination to be used as the target for error messages produced by CICSOPT and IDMSTRUE. The default destination-name is CSMT. Use another destination if you want to rout CA-IDMS error messages to another CICS destination. The DCT entry should be defined with a logical record length of at least 130 characters.

PLT=YES/NO

Indicates how CICSOPT starts up. YES indicates that CICSOPT can start up as a PLT-invoked program. NO indicates CICSOPT always starts up as a user task once CICS start up is complete.

TIMEOUT

Indicates how long a block of storage will be held after termination of a CICS task that utilizes CA-IDMS services. This storage block is acquired with a CICS GETMAIN and is initialzed when a terminal makes a request for IDMS services.

Valid values are 1 through 10, 0, and IMMEDIATE. A value between 1 and 10 indicates that the storage will be held for the specified number of minutes. If another CA-IDMS request is made within that time, the same block of storage will be utilized. If the storage is not used within that time, it will be freed. The next request for CA-IDMS services will result in the acquistion and initialization of a new storage block.

A value of 0 (the default) indicates that once a block of storage is acquired for a particular CICS terminal that storage will not be freed until either of these things happen:

- The user signs off the terminal
- A task using that storage block aborts

A value of IMMEDIATE indicates that the storage will be freed at task termination of the task that acquired the storage. If IMMEDIATE is coded, a new storage block is acquired and initialized for each CICS task that requests CA-IDMS services.

Notes:

The following considerations apply to the TIMEOUT parmeter:

1. TIMEOUT=IMMEDIATE must be specified for any IDMSINTC module that runs in a CICS Application Owning Region (AOR) and which is invoked from terminals in a separate Terminal Owning Region (TOR).
2. Since several thousand bytes are held by each terminal, a non-IMMEDIATE value may result in a large amount of storage being held for terminals with no active tasks.
3. A TIMEOUT value of 1 through 10 MUST NOT be coded if any application program using IDMSINTC services also uses READQ TS ... with the SET option.
4. If the INTCID parameter and/or the NTID parameter of the #UCFCICZ macro are not coded correctly or if the UCFCICZ exit is not driven for some other reason, the IDMS storage block may not be freed when a CICS user is logged

off a terminal. The next user who signs on that terminal may acquire that storage block. This can result in S019 or other S0nn abends or other application problems. If such problems are encountered, use TIMEOUT=IMMEDIATE.

TRANSID=

Identifies the transaction coded in the program control table (PCT) as invoking CICSOPT. Task-code must be the name of a task defined in the PCT table.

►► For more information about the PCT table, see *CA-IDMS System Operations*.

OPTIXIT=YES/NO

Indicates whether CICS transactions can modify the IDMSOPTI structure dynamically so that only the task thread is affected by the changes. YES indicates that the IDMSOPTI structure can be modified dynamically.

CICSOPT copies the static IDMSOPTI structure into dynamic storage and passes it to the user routine, which may alter it based on site-specific rules.

►► For more information about the OPTIXIT user exit, refer to *CA-IDMS System Operations*.

DICTNAM=

Sets the default SQL catalog. In general, existing native DML will not behave in the desired manner when you use this parameter.

DICTNOD=

Sets the default SQL catalog. In general, existing native DML will not behave in the desired manner when you use this parameter.

Assembling CICSOPTS and link editing IDMSINTC: JCL to assemble CICSOPTS and link edit the IDMSINTC module appears below for the OS/390 environment.

Note: The following JCL does not use SMP/E. For examples of how to apply a modification to a CA-IDMS load library using SMP/E, see the SAMPJCL library delivered with the CA-IDMS installation tape.

OS/390 CICSOPTS assembly and IDMSINTC link edit

```
//*-------------------------------------------------------------------
//*            ASSEMBLER ASMA90 JOB STREAM
//*-------------------------------------------------------------------
//ASMSTEP EXEC PGM=ASMA90,
//        PARM='ALIGN,XREF,PUNCH,NODECK',
//        REGION=2048K
//SYSLIB       DD  DSN=idms.maclib,DISP=SHR
//             DD  DSN=idms.srclib,DISP=SHR
//             DD  DSN=cics.maclib,DISP=SHR
//             DD  DSN=os390.maclib,DISP=SHR
//SYSUT1       DD DSN=&&SYSUT1,UNIT=VIO,SPACE=(1700,(600,100))
//SYSUT2       DD DSN=&&SYSUT2,UNIT=VIO,SPACE=(1700,(600,100))
//SYSUT3       DD DSN=&&SYSUT3,UNIT=VIO,SPACE=(1700,(600,100))
//SYSPRINT     DD  SYSOUT=*
//SYSPUNCH     DD  DSN=&&OBJECT,
                  DISP=(NEW,PASS),
                  UNIT=SYSDA,
                  SPACE=(80,(500,1000))
//SYSIN        DD  *
   CICSOPT input parameters
  END
//*-------------------------------------------------------------------
//*            LINK IEWL
//*-------------------------------------------------------------------
//LINK    EXEC PGM=IEWL,
//        PARM='LET,LIST,XREF,RENT',
//        REGION=128K,
//        COND=(8,LT,ASMSTEP)
//SYSLMOD      DD  DSN=idms.loadlib,DISP=SHR
//SYSPRINT     DD  SYSOUT=*
//SYSUT1       DD  DSN=&&SYSUT1,
//             UNIT=SYSDA,
//             SPACE=(6400,(80)),
//             DISP=(NEW,PASS)
//IN1          DD  DSN=idms.distload,DISP=SHR
//IN2          DD  DSN=cics.loadlib,DISP=SHR
//IN3          DD  DSN=&&OBJECT,DISP=(OLD,DELETE)
//SYSLIN       DD  DDNAME=SYSIN
//SYSIN        DD  *
 ORDER DFHEAI
 INCLUDE IN2(DFHEAI)
 INCLUDE IN3
 INCLUDE IN1(INTC0410)
 INCLUDE IN1(IDMSTRUE)
 INCLUDE IN1(IDMS)
 INCLUDE IN2(DFHEAIO)
 INCLUDE IN1(ESCD0410) ; Remove if DL/I Transparency is not desired
 INCLUDE IN1(ESCT0410) ; Remove if TOTAL Transparency is not desired
```

```
 INCLUDE IN1(ESCV0410) ; Remove if VSAM Transparency is not desired
 ENTRY STARTUP
 MODE AMODE(31),RMODE(24)
 NAME IDMSINTC(R)
/*
```

Item	Description
cics.loadlib	data set name of the CICS load library
cics.maclib	data set name of the CICS macro library
idms.distload	data set name of the CA-IDMS SMP/E distribution load library
idms.loadlib	data set name of the CA-IDMS load library
idms.maclib	data set name of the CA-IDMS macro library
idms.srclib	data set name of the CA-IDMS source library
os390.maclib	data set name of the OS/390 system macro library

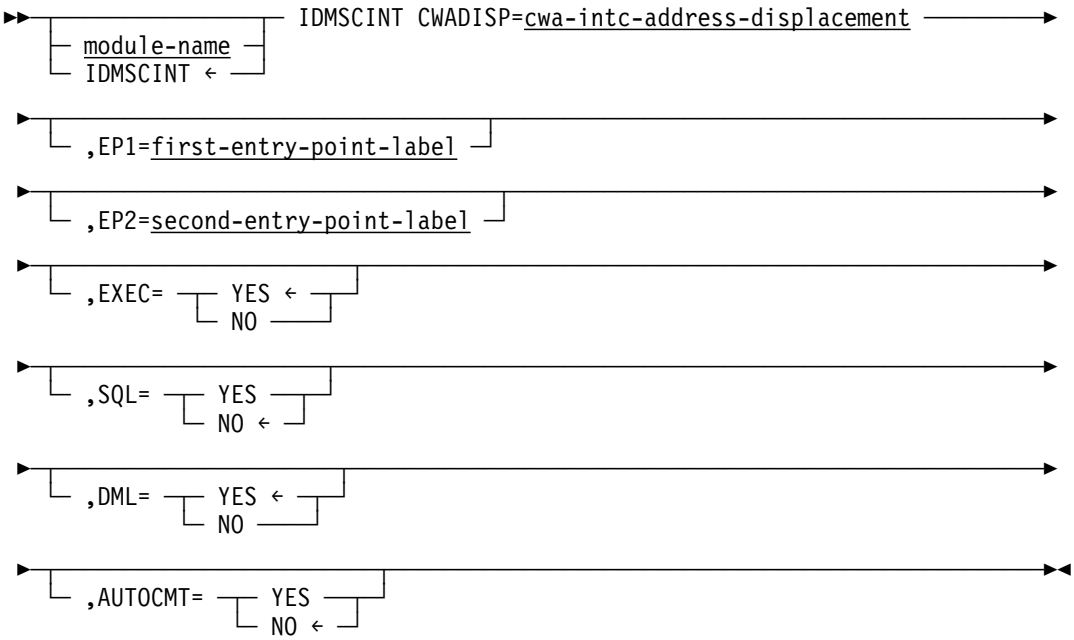
E.2 IDMSCINT

You generate IDMSCINT at DC/UCF installation time. IDMSCINT must be link edited with each CICS user program that accesses DC/UCF (including Assembler modules). IDMSCINT passes the entry point address to IDMSINTC from the CWA and passes control to IDMSINTC. The module generated by the IDMSCINT macro is fully reentrant.

▸▸ For more information about run-time events, see *CA-IDMS System Operations*.

To prepare an IDMSCINT macro, use the syntax shown below. If both the command-level and the macro level interfaces are being used, prepare separate IDMSCINT modules with distinct names for each interface.

IDMSCINT syntax



IDMSCINT parameters

module-name
Identifies the CSECT name of the generated module.

CWADISP=
Identifies the displacement within the CICS CWA of a fullword that holds the address of the IDMSINTC module. For *cwa-intcddress-displacement*, specify the same value given to the CWADISP operand of the IDMSINTC macro.

EP1=
Identifies the name of the first entry point in the generated module. For *first-entry-point-label*, specify the name of the first entry point. The default value

is either the value given to *module-name* above or IDMSCINT if *module-name* is not specified.

EP2

Identifies the name of the second entry point in the generated module. For *second-entry-point-label*, specify the name of the second entry point.

Considerations:

- A second entry point is present only if the EXEC parameter described below specifies NO. In this case, if EP2 is not specified, the default value is IDMSINC2.
- If the EXEC parameter is coded as YES, then no second entry point is generated, and EP2 should not be specified.

EXEC=YES/NO

Indicates whether the IDMSCINT module being generated is to be used with the CICS command-level interface.

YES

Specify YES to indicate that the command-level interface is in use. YES is the default.

If EXEC=YES, then name the module IDMSCINT to allow it to be autolinked with applications.

NO

Specify NO to indicate that the command-level interface is not in use. EXEC=NO must be specified if installing the CA-IDMS DMS interface.

If EXEC=NO, then name the module IDMSINC1 to allow the use of autolink.

SQL=YES/NO

Determines whether the program linked with IDMSCINT can (YES) or cannot (NO) issue SQL DML statements. NO is the default.

DML=YES/NO

Determines whether the program linked with IDMSCINT can (YES) or cannot (NO) issue non-SQL DML statements. YES is the default.

AUTOCMT=YES/NO

Specifies whether IDMSTRUE intercepts a CICS SYNCPOINT or SYNCPOINT ROLLBACK command issued by an application and translates it to an IDMS FINISH TASK or ROLLBACK TASK command, respectively. YES instructs IDMSTRUE to intercept the SYNCPOINT and SYNCPOINT ROLLBACK commands. At normal task termination IDMSTRUE issues a FINISH TASK command on behalf of the application; for abnormal termination, it issues a ROLLBACK TASK command. This option applies to applications coded for Release 12.0 and above.

NO, the default, means IDMSTRUE does not intercept the SYNCPOINT commands. NO supports old applications that do not desire to have CICS SYNCPOINT requests intercepted and serviced on their behalf. Such applications must COMMIT or FINISH the database transaction before terminating; otherwise,

CA-IDMS rolls back updates processed between the last COMMIT and task termination.

Appendix F. CA-Culprit Profile Options

F.1 Keywords and Operands F-4
F.1.1 Block/Track Option F-4
F.1.2 CALC Key Sign Option F-4
F.1.3 Headers Option F-4
F.1.4 Assembly Date Option F-4
F.1.5 Buffer Size Option F-5
F.1.6 Lines Per Page Option F-5
F.1.7 Date Stamp Option F-5
F.1.8 Error Options F-5
F.1.9 Report Error Level Option F-6
F.1.10 Hexadecimal Dump Option F-6
F.1.11 IDMS Buffer Size Option F-6
F.1.12 DDNAME Modification Option F-6
F.1.13 Report Lines Per Page Option F-7
F.1.14 Line Size Option F-7
F.1.15 Numeric Editing Option F-7
F.1.16 Operating System Option F-7
F.1.17 Tape Records/Block Option F-7
F.1.18 Return Codes Option F-8
F.1.19 Relocating Loader Option F-8
F.1.20 Schema Name Option F-8
F.1.21 Repeat First Page Option F-8
F.1.22 SPIE/STXIT Routine Option F-8
F.1.23 File Characteristics Option F-9
F.1.24 Time Stamp Option F-10
F.1.25 Separator Character Option F-10
F.1.26 Source Library Option F-10
F.1.27 CA-Panvalet File Option F-10

This appendix provides information about the keywords and operands for CA-Culprit profile options. The CA-Culprit profile options determine the report format, date format and other site-specific characteristics. For more information about the profile options or overriding the options at runtime, see the *CA-Culprit Reference Guide*.

Note: During installation, these options can be specified in the VARBLIST variables GJCPRF1 through GJCPRF9.

F.1 Keywords and Operands

F.1.1 Block/Track Option

This option is for VSE/ESA systems only and specifies the number of blocks per track for CA-Culprit work files written to disk:

```
BT={ 1 ←
     2
     4
     8}
```

F.1.2 CALC Key Sign Option

This option indicates if IDMS CALC keys are stored with an F or C sign:

```
CALSIN={F ←
       C }
```

- F — Specifies packed decimal IDMS CALC keys are stored with an F sign.
- C — Specifies packed decimal IDMS CALC keys contain a C sign.

F.1.3 Headers Option

This option indicates if headers are forced:

```
CCH={ N ←
      A
      G }
```

- N — Specifies no headers are forced when a detail or total lines specifies a carriage control of 1. Headers print only when the specified lines per page for the report is exceeded or when a control break occurs for a sort field with a break code on 1.
- A — Specifies headers are always printed whenever a detail line or total line specify a carriage control of 1.
- G — Specifies headers are forced by a line printing with a carriage control of 1 only if it is a total line and the totals being printed are grand totals.

F.1.4 Assembly Date Option

This option indicates the PROFILE assembly date:

```
DATE=mm/dd/yy
```

mm/dd/yy — Specifies the date the PROFILE CSECT was assembled.

This option should always be specified for diagnostic purposes.

F.1.5 Buffer Size Option

This option specifies report buffer size:

```
DICTB=buffer-size-n
```

buffer-size-n — Specifies the record size that is used for reports generated for the Data Dictionary Reporter and for IDMS-DC reports.

Default: 1000 Maximum: 32767

F.1.6 Lines Per Page Option

This option specifies the number of lines per page on each diagnostic list:

```
DLLP=lines-per-page-n
```

lines-per-page-n — Specifies the number of lines per page that prints in the CA-Culprit diagnostic listings. This option does not affect the number of lines per page in output reports.

Default: 55

F.1.7 Date Stamp Option

This option specifies the format of the date stamp that appears on CA-Culprit listings and user output reports:

```
DS= { A
      E
      C )
```

Note: Refer to the *CA-Culprit Reference Guide* for further information on date stamp.

F.1.8 Error Options

These options specify the number of errors to be reported and the manner in which errors are processed. If this option is used in the MACRO, the option must be entered as indicated in the syntax shown below:

```
{ EE=
  IE=
  ME=      (error-count-n, return-code-n {I  {D )
  OE=                                    A   N}
  PE=                                    S}
  SE=  }
```

Note: Refer to the *CA-Culprit Reference Guide* for further information on error options.

F.1.9 Report Error Level Option

This option specifies the error codes under which CA-Culprit will read and report on the user's input files:

```
EX= { W
      E
      N }
```

Note: Refer to the *CA-Culprit Reference Guide* for further information on this option.

F.1.10 Hexadecimal Dump Option

This option specifies the format in which hexadecimal dumps issued by the extended error handling facility are to appear:

```
HD= { V
      H }
```

Note: Refer to the *CA-Culprit Reference Guide* for further information on this option.

F.1.11 IDMS Buffer Size Option

This option specifies the buffer size for IDMS access:

```
IDMSB=record-size-n
```

record-size-n — Specifies the buffer size reserved for IDMS database accesses when the record size field is omitted from the INPUT parameter.

Default: 1000 Maximum: 32767

F.1.12 DDNAME Modification Option

This option, which is for OS/390 only, is used to change CA-Culprit step ddnames:

```
{ IN0=    (ddname,00,' ',00)
  IN4= }
```

IN0/IN4 specifies the CA-Culprit step whose ddname is changed:

- IN0 - CULP0 step
- IN4 - CULP4 step
- *ddname* — Specifies the ddname to which the step is to be changed. The default is SYSIN.

F.1.13 Report Lines Per Page Option

This option specifies the maximum number of lines per page for all user reports:

```
LP=lines-per-page-n
```

Note: Refer to the *CA-Culprit Reference Guide* for further information on this option.

F.1.14 Line Size Option

This option specifies the line size for all printed reports:

```
LS=line-count-n
```

Note: Refer to the *CA-Culprit Reference Guide* for further information on this option.

F.1.15 Numeric Editing Option

This option indicates American or European numeric editing:

```
MONEYED= { A
           E }
```

- A — Specifies the default numeric editing is to be done in the American fashion, that is, decimal digits separated from integer digits with a period, and integer digits separated into groups of three with commas, for example, 1,000.52.
- E — Specifies European editing, that is, decimal digits separated from integer digits with a comma, and integer digits separated into groups of three with a period, for example, 1.000,52.

F.1.16 Operating System Option

This option specifies the operating system:

```
OPSYS= { O
         D }
```

- O — Specifies the OS/390 operating system.
- D — Specifies the VSE/ESA operating system.

Note: Because the default is specified at installation, you don't usually code this option.

F.1.17 Tape Records/Block Option

The option, which applies to VSE/ESA systems only, specifies the size of CA-Culprit work files written to tape:

```
RB= { 8 ←
      4
      2
      1 }
```

RB= (records/blocks) — Specifies the size of CA-Culprit work files written to tape, expressed in 780-byte records per block.

F.1.18 Return Codes Option

This option specifies the return codes associated with the four levels of CA-Culprit messages:

```
RC=(return-code-n..)
```

Note: Refer to the *CA-Culprit Reference Guide* for further information on this option.

F.1.19 Relocating Loader Option

This option, which is for VSE/ESA systems only, indicates if a relocating loader is present:

```
RELO= { Y ←
        N }
```

- Y — Indicates the operating system has a relocating loader.
- N — Indicates the operating system does not have a relocating loader.

F.1.20 Schema Name Option

This option indicates if a schema name is required:

```
SCHMREQ= { Y
           N ←}
```

- Y — Indicates the schema name is required. The schema version remains optional.
- N — Indicates the user can omit coding the schema name when entering the SS= operand on the INPUT parameter for access of an IDMS database.

F.1.21 Repeat First Page Option

This option specifies if the first page of printed output on special forms is to be repeated.

```
SF= { N
      E }
```

Note: Refer to the *CA-Culprit Reference Guide* for further information on this option.

F.1.22 SPIE/STXIT Routine Option

This option indicates if the SPIE/STXIT routine is enabled:

```
SPIE= { Y ←
        N }
```

- Y — Specifies CA-Culprit extended error handling SPIE/STXIT routine is enabled.

- N — Specifies the SPIE/STXIT routine is disabled. Program check errors that normally are trapped by CA-Culprit result in immediate abnormal termination of the job.

F.1.23 File Characteristics Option

```
S work-file-n=(filename, logical-unit-n, file-type, dtf-code)  -
```

- *work-file-n* — Specifies the SYS number of a CA-Culprit work, input, or key file whose default characteristics are to be overridden.
- Valid codes:
  - 2
  - 3
  - 4
  - 5
  - 6
  - 7
  - 8
  - 10

  These codes correspond to the name SYS002, SYS003, and so on.

  Note: Refer to the *CA-Culprit Reference Guide* for further information on SYS numbers.
- *filename* — Specifies the filename (VSE/ESA), ddname (OS/390), or linkname (BS2000/OSD) to be used for this file.
- *logical-unit-n* — Specifies the logical unit number associated with this file. Valid values are from 1 through 256.

  OS/390 users must code a value to satisfy the MACRO, but CA-Culprit ignores the value.
- *file-type* — Specifies the type of file. Valid codes are T (tape), D (disk), and blank enclosed in single quotation marks (indicates a LUB/PUB lookup).

  OS/390 users must code a value to satisfy the MACRO, but CA-Culprit ignores the value.
- *dtf-code* — A 2-character hexadecimal code representing the one-byte DTF code of the device for the file. FF (the default) indicates that a LUB/PUB lookup is performed.

  OS/390 users must code a value to satisfy the MACRO, but CA-Culprit ignores the value.

F.1.24 Time Stamp Option

This option indicates if a time stamp is to appear on parameter listings and user reports. If this option is specified, all three values must be included, as indicated by the following syntax:

```
TS=( { Y   }, { Y   }, { Y   } )
       N ←      N ←      N ←
```

Note: Refer to the *CA-Culprit Reference Guide* for further information on this option.

F.1.25 Separator Character Option

This option specifies the hour/minute separator:

```
TSEP= { : ←
        / }
```

- : — Indicates a colon separates hours from minutes on a time stamp.
- / — Indicates a slash separates hours from minutes on a time stamp.

F.1.26 Source Library Option

This option specifies the source library that CA-Culprit will use for =COPY and =MACRO parameters:

```
PARMLIB=source-library-a
```

For source-library-a, specify one of the following values:

- *STANDARD* — Indicates the source library is a source statement library (VSE/ESA) or a partitioned dataset (OS/390).
- *PANVALET* — Indicates the source library is a CA-Panvalet library.
- *LIBRARIAN2* — Indicates the source library is a LIBRARIAN (Release 2.0) library.
- *LIBR30* — Indicates the source library is a LIBRARIAN (Release 3.0) library.
- *LIBRARIAN3* — Indicates the source library is a LIBRARIAN (Release 3.1 and above) library.

F.1.27 CA-Panvalet File Option

This option, which applies to VSE/ESA systems only, identifies the logical unit number and devices that CA-Culprit uses for accessing a CA-Panvalet library with =COPY and =MACRO parameters:

```
PANFILE='logical-unit-number=device-type1-a,device-type2-a'
```

Default: PANFILE='SYS006=333-,3330'

The single quotation marks are required.

Appendix G. CA-IDMS Tools Runtime Options

G.1 CA-IDMS/ADS Alive Runtime Parameters G-4
G.2 CA-IDMS/Database Extractor Runtime Parameters G-5
G.3 CA-IDMS/Dictionary Migrator Runtime Parameters G-7
G.4 CA-IDMS/Dictionary Migrator Assistant Runtime Parameters G-22
G.5 CA-IDMS/Dictionary Module Editor Runtime Parameters G-23
G.6 CA-IDMS/Dictionary Query Facility Runtime Parameters G-25
G.7 CA-IDMS/DML Online Runtime Parameters G-26
G.8 CA-IDMS/Enforcer Runtime Parameters G-37
G.9 CA-IDMS/Master Key Runtime Parameters G-38
G.10 CA-IDMS/Online Log Display Runtime Parameters G-39
G.11 CA-IDMS/SASO Runtime Parameters G-40
G.12 General Sort Runtime Parameters G-41

This appendix describes the CA-IDMS Tools runtime parameters. These parameters are supplied with default values and can be modified at installation time by changing the VARBLIST member.

These runtime parameters can also be modified after initial installation by changing selected macro parameters in a particular customization module (xxxTPARM), and re-assembling and re-linking that module.

Note: The installation procedure defines, initializes, and loads a dictionary with various product modules. This is the dictionary that is referred to by the HLPDICT and HLPNODE parameters that appear in most of the xxxTPARM modules.

See SAMPJCL member UMOD1 for a USERMOD example of how to change xxxTPARM values.

G.1 CA-IDMS/ADS Alive Runtime Parameters

```
*--------------------------------------------------------------------
* CA-IDMS/ADS-ALIVE RUNTIME PARAMETERS
*--------------------------------------------------------------------
*
*USGTPARM THIS IS THE INSTALLATION TAILORING MACRO USED BY THE
*USG SYSTEM TO PROVIDE RUN-TIME VALUES.
*OPERANDS:
*                 USGTSK='1-8 CHAR'    TASK USED TO INVOKE USG.
*                 HLPDICT='1-8 CHAR'   ALTERNATE DICTIONARY USED
*                                      FOR GSIHELP.
*                 HLPNODE='1-8 CHAR'   ALTERNATE NODE USED
*                                      FOR GSIHELP.
*
*                 HLPVERS=INTEGER      VERSION NUMBER OF HELP
*                                      MODULES.
*                 PCHOFF= INTEGER      OFFSET FOR IMPLANT
*
*                 SWEEP= (Y OR N)      YES OR NO - AREA SWEEP FOR
*                                      DIALOG WILD CARDS
*                 AUTO=  (Y OR N)      YES OR NO - NON-INTERRUPT
*                                      MODE ALLOWED
*
*                 QKEEP=INTEGER        NUMBER OF DAYS TO RETAIN
*                                      DEBUGQUEUE RECORDS
*
*
*                 PROKEEP=INTEGER      NUMBER OF DAYS TO RETAIN
*                                      ADSALIVE
*                                      PROFILE QUEUE RECORDS
*                                      MUST BE NUMERIC INTEGER
*                                      BETWEEN 0 AND 255
*
*                 DICTDEF= (D OR P)    D = DICTNAME WILL BE FROM
*                                          DEFAULT DICTNAME
*                                      P = DICTNAME WILL BE FROM
*                                          PROFILE.
*                                          (DEFAULT = P)
*                                      NOTE: FIRST TIME WILL
*                                      ALWAYS COME FROM DEFAULT
*                                      DICTNAME
*
*
*ASSEMBLED VALUES AT INSTALLATION:
*       USGTPARM USGTSK='ADSALIVE',
*                HLPDICT='        ',
*                HLPNODE='        ',
*                HLPVERS=1,
*                PCHOFF=3800,
*                SWEEP=Y,
*                AUTO=Y,
*                QKEEP=3,
*                PROKEEP=255,
*                DICTDEF=P
*--------------------------------------------------------------------
```

G.2 CA-IDMS/Database Extractor Runtime Parameters

```
*--------------------------------------------------------------------
*CA-IDMS/DATABASE EXTRACTOR RUNTIME PARAMETERS
*--------------------------------------------------------------------
*            MODIFY PRODUCT TUNING PARAMETERS
*            ________________________________
*USVTPARM ── THIS MEMBER IS USED TO SPECIFY THE RUNTIME VALUES TO
*            BE USED AS INPUT TO THE INSTALLATION TAILORING MACRO,
*            USVCPARM.
*
*            RUNTIME VARIABLES
*            _________________
*            TASK='1-8 CHAR'      TASK USED TO INVOKE DBX.
*
*            HLPDICT='1-8 CHAR'   DICTNAME OF DICTIONARY INTO WHICH
*                                 USVTUTOR MODULES WERE ADDED.  NULL FOR
*                                 DEFAULT DICTIONARY.
*
*            HLPNODE='1-8 CHAR'   DICTNODE FOR "HLPDICT" - NULL IF NO DDS.
*
*            HLPVERS=INTEGER      VERSION NUMBER AT WHICH USVTUTOR MODULES
*                                 WERE ADDED: MUST BE   1 - 9999
*
*            STKENTS=INTEGER      # OF 8 BYTE ENTRIES TO ALLOCATE FOR DBX
*                                 SET STACK: MUST BE 30 - 1000
*                                 THE NUMBER OF SETS THAT WILL BE TRAVERSED
*                                 IN YOUR EXTRACT PATH BEGINNING AT THE
*                                 DATABASE ENTRY POINT. A SAFE NUMBER WOULD
*                                 BE ONE FOR EACH SET IN YOUR SUBSCHEMA. FOR
*                                 EXAMPLE, 200 IS A SUITABLE VALUE FOR IDMSNWKA.
*
*            COPY='1-8 CHAR'      WHO A USER CAN COPY OTHER JCL MEMBERS
*                                 AND SPECIFICATIONS FROM:
*                                   'USER' ──── FROM ONLY HIM/HERSELF;
*                                   'DBXADMIN' - FROM HIM/HERSELF PLUS ANY
*                                                GLOBAL MEMBERS UNDER THE
*                                                'DBXADMIN' USER-ID;
*                                   'ANYONE' ─── FROM ANYONE ON THE DBX
*                                                DATABASE.
*
*            RETSEQ=Y|YES|N|NO    DEFAULT 'RETAIN PHYSICAL SEQUENCE OF
*                                 MEMBER RECORDS IN THE SET?' VALUE ON
*                                 THE RECORD LEVEL SELECTION CRITERIA SCREEN.
*
*            XRECURO=Y|YES|N|NO   DEFAULT 'EXTRACT ALL OWNERS FOR
*                                 EXTRACTED RECURSIVE RECORDS?' VALUE ON
*                                 THE RECORD LEVEL SELECTION CRITERIA SCREEN.
*
*            BGINMID=Y|YES|N|NO   DEFAULT 'BEGIN VIEWING/EDITING IN THE
*                                 MIDDLE OF A PATH DEFINITION' VALUE ON THE
*                                 SPECIFY DATABASE EXTRACT SPECIFICATION SCREEN.
*
*            NLYZ008=W|WARNING|   HAVE MESSAGE NLYZ008 AS A WARNING OR
*                    E|ERROR      ERROR MESSAGE.  NLYZ008 IS DISPLAYED
*                                 AT EXTRACT TIME WHEN A MANDATORY MEMBER
*                                 IS BEING EXTRACTED WITHOUT ITS OWNER.
*                                 AN ERROR MESSAGE PREVENTS THE
*                                 SPECIFICATION FROM BEING USED.
*--------------------------------------------------------------------
*  DEFAULT VALUES AS SUPPLIED WITH INSTALLATION:
*       USVCPARM TASK=DBX,
*                HLPDICT=,          NULL
*                HLPNODE=,          NULL
*                HLPVERS=1,
*                STKENTS=50,
*                COPY=ANYONE,
*                RETSEQ=YES,
*                XRECURO=YES,
*                BGINMID=YES,
*                NLYZ008=WARNING
*--------------------------------------------------------------------
```

G.3 CA-IDMS/Dictionary Migrator Runtime Parameters

```
*--------------------------------------------------------------------
* CA-IDMS/DICTIONARY MIGRATOR RUNTIME PARAMETERS
*--------------------------------------------------------------------
*  DICTIONARY MIGRATOR
*  RELEASE 15.0
*  PRODUCT CUSTOMIZATION INSTRUCTIONS
*
*  THE FOLLOWING INSTRUCTIONS EXPLAIN WHAT CUSTOMIZATION OPTIONS ARE
*  AVAILABLE FOR DICTIONARY MIGRATOR AND HOW TO IMPLEMENT ANY OPTION
*  CHOSEN.
*
*  NOTE: THESE CUSTOMIZATION OPTIONS ARE NOT REQUIRED FOR THE PROPER
*  EXECUTION OF DICTIONARY MIGRATOR.  IF THE MODULE PROVIDED ON THE
*  INSTALLATION TAPE OR ALL DEFAULT VALUES ARE USED, DICTIONARY
*  MIGRATOR WILL EXECUTE A CORRECT MIGRATION FOR THE ENTITY(S) NAMED IN
*  THE PARAMETER STATEMENTS.  THESE OPTIONS ARE PROVIDED FOR USERS
*  WHOSE SHOP STANDARDS MANDATE SOME DEVIATION FROM THE BASIC MIGRATION
*  STRATEGY.
*
*  GENERAL:
*
*  THE CUSTOMIZATION OPTIONS FOR DICTIONARY MIGRATOR ARE FOUND IN THIS
*  MODULE.  A VERSION OF THIS MODULE WITH ALL DEFAULT VALUES SPECIFIED
*  IS PROVIDED IN LOAD MODULE FORM IN THE INSTALLATION.  IN THIS
*  MEMBER, EACH OPTION IS LISTED WITH ITS DEFAULT VALUE.  TO CHANGE AN
*  OPTION, CHANGE THE VALUE OF THE RELEVANT PARAMETER.  SMP/E WILL
*  ASSEMBLE AND LINK THIS MODULE.  THE ONLY VALID VALUES FOR ANY
*  PARAMETER IN USMTPARM ARE LISTED IN THIS SUPPLEMENT;  ANY OTHER
*  VALUE WILL RESULT IN A LEVEL 8 ERROR DURING ASSEMBLY.
*
*  ASSEMBLY AND LINKAGE:
*  ANY LEVEL OF IBM ASSEMBLER AND LINKAGE EDITOR CAN BE USED TO CREATE
*  THE USMTPARM MODULE.
*--------------------------------------------------------------------
```

```
*-----------------------------------------------------------------------
*  A NOTE ON NUMBERING IN THESE INSTRUCTIONS:
*
*  THE NUMBERS WHICH PRECEDE THE OPTIONS LISTED IN THESE INSTRUCTIONS
*  CAN ALSO BE USED AS THE OFFSET TO THE RELEVANT BYTE WITHIN THE
*  LOAD MODULE.  THIS IS USEFUL WHEN VERIFYING WHICH OPTIONS ARE IN
*  EFFECT.
*
*  WHEN AN OPTION HAS NO NUMBER PRECEDING IT, THE OPTION DOES NOT
*  AFFECT A SINGLE BYTE, BUT RATHER AFFECTS THE VALUES OF SEVERAL
*  BYTES. XUDNREF AND XUDNXRT ARE THE PRIMARY OCCURRENCES OF SUCH
*  "GROUP" OPTIONS.
*
*-----------------------------------------------------------------------
*  INDIVIDUAL OPTIONS
*-----------------------------------------------------------------------
*  1. XPICOVR (EXCLUDE PICTURE OVERRIDES)
*  -  PURPOSE: PRODUCE ADD RECORD SYNTAX WITHOUT PICTURE OVERRIDE
*     CLAUSES FOR RECORD ELEMENTS.
*  -  DEFAULT: RECORD SYNTAX IS CREATED INCLUDING PICTURE OVERRIDE
*     CLAUSES FOR ALL RECORD ELEMENTS.
*  -  TO INVOKE THIS OPTION, CODE: XPICOVR=Y
*  -  TO USE THE DEFAULT, CODE: XPICOVR=N
*  -  COMMENTS: PICTURE OVERRIDES ARE NEEDED FOR CORRECT MIGRATION ANY
*     TIME THAT THE ELEMENT PICTURE AND THE PICTURE AS USED IN THE
*     RECORD ARE NOT IDENTICAL.  USING THE DEFAULT VALUE INSURES THAT
*     THE RECORD ADDED TO THE OBJECT DICTIONARY WILL BE IDENTICAL TO
*     THE SOURCE DICTIONARY WITHOUT AN ADDED STEP OF MANUAL
*     VERIFICATION.
*-----------------------------------------------------------------------
*
*  2.      XSUBEL   (EXCLUDE SUBORDINATE ELEMENTS)
*  -   PURPOSE: PRODUCE ADD RECORD SYNTAX WITHOUT SUBORDINATE ELEMENT
*      IS CLAUSE.
*  -   DEFAULT: RECORD SYNTAX IS CREATED INCLUDING SUBORDINATE
*      ELEMENTS CLAUSES FOR ALL RECORD ELEMENTS.
*  -   TO INVOKE THIS OPTION, CODE: EXSUBEL=Y
*  -   TO USE THE DEFAULT, CODE: EXSUBEL=N
*  -   COMMENTS: THE SUBORDINATE ELEMENT CLAUSE PROVIDES MORE COMPLETE
*      DOCUMENTATION OF THE STRUCTURE OF THE RECORD AND ALSO VERIFIES
*      THAT GROUP ELEMENTS DEFINITIONS ARE IDENTICAL TO THE USE OF
*      THE GROUP ELEMENT WITHIN THE RECORD.
*-----------------------------------------------------------------------
```

```
*-------------------------------------------------------------------
*
*  3.      MAPDCMP   (MAP DECOMPILE)
*  -   PURPOSE: USE THE BATCH MAPPING FACILITY OPTION PROCESS=DECOMPILE
*      WHEN PRODUCING MAP (RHDCUPD) SYNTAX.
*  -   DEFAULT: MAP SYNTAX IS CREATED USING THE PROCESS=TERSE UNLESS
*      EITHER 1) NEWVERSION OR 2) CHANGEONLY AND RUN=AUDIT ARE ELECTED,
*      IN WHICH CASE PROCESS=DECOMPILE IS AUTOMATICALLY USED.
*  -   TO INVOKE THIS OPTION, CODE: MAPDCMP=Y
*  -   TO USE THE DEFAULT, CODE: MAPDCMP=N
*  -   COMMENTS: PROCESS=TERSE PRODUCES MAP SYNTAX WHICH IS MUCH MORE
*      CONCISE THAN PROCESS=DECOMPILE.  NORMALLY, THAT OPTION SHOULD BE
*      USED.  HOWEVER, PROCESS=TERSE OMITS ALL PARAMETERS WHERE THE
*      VALUE IS THE DEFAULT, THUS ERRORS MAY BE INTRODUCED WHEN
*      MIGRATING BETWEEN UNLIKE ENVIRONMENTS.  LIKEWISE, MIGRATION
*      BETWEEN DIFFERENT RELEASE LEVELS OF IDMS MAY BE UNPREDICTABLE
*      USING PROCESS=TERSE.
*-------------------------------------------------------------------
*
*  4.      SHARRDY   (READY IN SHARED UPDATE)
*  5.      EXCLRDY   (READY IN EXCLUSIVE UPDATE)
*  -   PURPOSE: DEFINE THE USAGE MODE TO BE USED IN THE UPLOAD STEPS
*      OF DICTIONARY MIGRATOR.
*  -   DEFAULT: DICTIONARY AREAS ARE READIED IN PROTECTED UPDATE.
*  -   SHARRDY AND EXCLRDY ARE MUTUALLY EXCLUSIVE, AT MOST, ONLY ONE
*      CAN BE CODED AS 'Y'.
*  -   TO READY IN PROTECTED UPDATE (DEFAULT): SHARRDY=N,EXCLRDY=N
*  -   TO READY IN SHARED UPDATE: SHARRDY=N,EXCLRDY=Y
*  -   TO READY IN EXCLUSIVE UPDATE: SHARRDY=N,EXCLRDY=Y
*  -   COMMENTS: REFER TO CA-IDMS PROGRAMMER'S GUIDE FOR AN OVERVIEW OF
*      USAGE MODES.  BECAUSE MIGRATION USUALLY INVOLVES UPDATES TO A
*      LARGE NUMBER OF DICTIONARY RECORDS,  PROTECTED UPDATE IS
*      RECOMMENDED.
*-------------------------------------------------------------------
*
*  6.      DFLTOFF   (DEFAULT IS OFF)
*  -   PURPOSE: SET OPTIONS FOR SESSIONS FOR THE UPLOAD STEPS
*      OF DICTIONARY MIGRATOR TO 'DEFALUT IS OFF'
*  -   DEFAULT: 'DEFAULT IS ON' IS USED.
*  -   TO INVOKE THIS OPTION, CODE: DFLTOFF=Y
*  -   TO USE THE DEFAULT, CODE: DFLTOFF=N
*  -   COMMENTS: THIS OPTION AFFECTS THE DISPOSITION OF ADD STATEMENTS
*      DURING THE UPLOAD STEPS.  WHEN THE DEFAULT IS USED, IF AN ADD
*      STATEMENT IS ENCOUNTERED FOR AN ENTITY OCCURRENCE ALREADY IN
*      THE DICTIONARY, THE ADD WILL BE CHANGED TO A MODIFY.  WITH THE
*      OPTION DFLTOFF=Y, THE ADD WILL BE TREATED AS AN ERROR, AND NO
*      UPDATE WILL OCCUR.
*  -   NOTE: DEFAULT IS OFF IS ALWAYS USED FOR RECORDS AS THE ADD
*      RECORD SYNTAX IS NOT COMPATIBLE WITH THE MODIFY COMMAND.
*-------------------------------------------------------------------
```

```
*
*  7.      PROGALL   (DISPLAY PROGRAM WITH ALL)
*  -   PURPOSE: CREATE DDDLPGM STATEMENTS IN WHICH PROGRAM ENTITIES
*      ARE DISPLAYED WITH ALL RELATIONSHIPS.
*  -   DEFAULT: PROGRAMS ARE DISPLAYED WITH A LIMITED RANGE OF
*      RELATIONSHIPS.
*  -   TO INVOKE THIS OPTION, CODE: PROGALL=Y
*  -   TO USE THE DEFAULT, CODE: PROGALL=N
*  -   COMMENTS: IN AN ADSO ENVIRONMENT, THE DDDLPGM PROGRAM STATEMENTS
*      ARE PRIMARILY FOR DOCUMENTATIONAL ENTRIES.  THE ADSOBGN STEP
*      ESTABLISHED A MAJORITY OF THE PROGRAM'S RELATIONSHIPS AND THEY
*      NEED NOT BE REPEATED IN THIS STEP.
*
*      IN OTHER ENVIRONMENTS, THIS OPTION CAN BE USEFUL IN ELIMINATING
*      THE NEED TO RERUN THE IDMSDMLX PREPROCESSOR TO REESTABLISH
*      PROGRAM STATISTICS.
*-----------------------------------------------------------------------
*  8.      XCLIST    (OMIT CLIST CREATION)
*  -   PURPOSE: ELIMINATE THE CREATION OF THE DCMT VARY NEW COPY CLIST
*  -   DEFAULT: THE CLIST IS CREATED
*  -   TO INVOKE THIS OPTION, CODE: XCLIST=Y
*  -   TO USE THE DEFAULT, CODE: XCLIST=N
*  -   COMMENTS: THE CLIST FEATURE OF DICTIONARY MIGRATOR IS A VERY
*      CONVENIENT METHOD OF IMMEDIATELY IMPLEMENTING A MIGRATED CHANGE.
*      IN SOME ENVIRONMENTS, HOWEVER, CHANGES ARE NOT SCHEDULED TO TAKE
*      EFFECT UNTIL THE SYSTEM IS RECYCLED.  IN SUCH CASES, THE CLIST
*      IS NOT NEEDED AND CAN BE OMITTED.
*-----------------------------------------------------------------------
*  THE NEXT 3 PARAMETERS ALL MAKE MODIFICATIONS TO THE STANDARD CLIST
*  FORMAT OF "DCMT VARY PROGRAM PROGRAM-NAME N C I".
*-----------------------------------------------------------------------
*  9.      XCLIMM    (OMIT IMMEDIATE OPTION FROM CLIST SYNTAX)
*  -   PURPOSE: CREATE THE CLIST SYNTAX WITHOUT IMMEDIATE OPTION
*  -   DEFAULT: THE CLIST IS CREATED WITH COMPLETE SYNTAX
*  -   TO INVOKE THIS OPTION, CODE: XCLIMM=Y
*  -   TO USE THE DEFAULT, CODE: XCLIMM=N
*  -   TO USE QUIESCE RATHER THAN IMMEDIATE, CODE: XCLIMM=Q
*  -   COMMENTS: THE IMMEDIATE OPTION IN VARY NEW COPY CAUSES THE
*      UPDATED LOAD MODULE TO BE LOADED IMMEDIATELY AFTER EXECUTION OF
*      THE COMMAND.  IF AN APPLICATION IS IN USE AT THIS TIME, SOME
*      UNEXPECTED RESULTS MAY OCCUR, INCLUDING ABNORMAL TERMINATION OF
*      USERS' SESSIONS.  OMITTING THE IMMEDIATE PARAMETER WILL CAUSE
*      THE UPDATE LOAD MODULE TO BE LOADED AT THE FIRST OPPORTUNITY
*      WHEN NO ONE IS USING THE MODULE.
*
*      THE QUIESCE OPTION FORMATS THE VARY PRO ... N C QUIESCE.  IN
*      THIS CASE ACTIVITY USING THE PROGRAM NAMED WILL BE QUIESCED.
*      WHEN NO ONE IS USING THE PROGRAM, A NEW COPY WILL BE LOADED.
*-----------------------------------------------------------------------
```

```
*-----------------------------------------------------------------------
*  10. XCLDBN        (OMIT DICTNAME FROM  CLIST SYNTAX)
*  -   PURPOSE: CREATE THE CLIST SYNTAX WITHOUT DICTNAME ENTRY
*  -   DEFAULT: THE CLIST IS CREATED WITH COMPLETE SYNTAX
*  -   TO INVOKE THIS OPTION, CODE: XCLDBN=Y
*  -   TO USE THE DEFAULT, CODE: XCLDBN=N
*  -   COMMENTS: THE DICTNAME ENTRY IN THE CLIST IS THE ONE NAMED AS
*      OBJECT DICTIONARY IN THE DICTIONARY MIGRATOR RUN.  HENCE, IT IS
*      ALSO THE DICTIONARY INTO WHICH THE CHANGED LOAD MODULES WERE
*      MOVED OR GENERATED.  OMITTING THIS PARAMETER ALLOWS ANOTHER SET
*      OF LOAD MODULES TO BE NEW COPIED, OR THE USE OF THE DCUF COMMAND
*      TO CONTROL THE DICTNAME USED.
*-----------------------------------------------------------------------
*  11. XCLVER        (OMIT VERSION FROM  CLIST SYNTAX)
*  -   PURPOSE: CREATE THE CLIST SYNTAX WITHOUT VERSION ENTRY
*  -   DEFAULT: THE CLIST IS CREATED WITH COMPLETE SYNTAX
*  -   TO INVOKE THIS OPTION, CODE: XCLVER=Y
*  -   TO USE THE DEFAULT, CODE: XCLVER=N
*  -   COMMENTS: THE VERSION ENTRY IN THE CLIST IS THE SPECIFIC VERSION
*      OF THE LOAD MODULE MOVED OR GENERATED IN THE OBJECT DICTIONARY.
*      IT IS NOT RECOMMENDED TO CHANGE THE VALUE OF THE OPTION.
*-----------------------------------------------------------------------
*
*  12. NOUDC         (EXCLUDE USER DEFINED COMMENTS)
*  -   PURPOSE: CREATE DDDLUPD AND DDDLPGM FILE SYNTAX WITHOUT ANY USER
*      DEFINED COMMENTS.
*  -   DEFAULT: THE SYNTAX IS CREATED INCLUDING ANY USER DEFINED
*      COMMENT TEXT THAT IS PRESENT IN THE SOURCE DICTIONARY.
*  -   TO INVOKE THIS OPTION, CODE: NOUDC=Y
*  -   TO USE THE DEFAULT, CODE: NOUDC=N
*  -   COMMENTS: USER DEFINED COMMENTS ARE COMMENTS WITH HEADERS OTHER
*      THAN THOSE DEFINED IN THE IDD AS DELIVERED.  THIS OPTION CREATES
*      UPLOAD SYNTAX WHICH DOES NOT INCLUDE THIS CATEGORY OF COMMENTS.
*      REFER TO TECHNICAL BULLETIN UM-9002-0004 FOR ADDITIONAL
*      INFORMATION REGARDING THE SUCCESSFUL MIGRATION OF USER DEFINED
*      COMMENTS.
*-----------------------------------------------------------------------
```

```
*-------------------------------------------------------------------
*      XUDNREF   (EXCLUDE ALL USER DEFINED NEST REFERENCES)
*
*  -   PURPOSE: CREATE UPLOAD SYNTAX WITHOUT ANY REFERENCES TO USER
*      DEFINED NESTS.
*  -   DEFAULT: THE SYNTAX IS CREATED INCLUDING ALL REFERENCES TO USER
*      DEFINED NESTS WHICH ARE PRESENT.
*  -   TO INVOKE THIS OPTION, CODE: XUDNREF=Y
*  -   TO USE THE DEFAULT, CODE: XUDNREF=N
*  -   COMMENTS:
*      THIS OPTION SERVES AS A GROUP ELECTION FOR ALL OF THE OPTIONS
*      RELATED TO INCLUDING REFERENCES TO USER DEFINED NESTS IN THE
*      SYNTAX CREATED.  IF THIS OPTION IS 'Y' ALL OF THE FLAGS BEGINNING
*      XUDNR ARE SET TO 'Y'.  IT IS NOT POSSIBLE TO OVERRIDE THIS
*      OPTION ON AN INDIVIDUAL ENTITY BASIS.  IF USER DEFINED NESTS
*      REFERENCES ARE DESIRED FOR SOME ENTITY TYPES, BUT NOT OTHERS,
*      CODE XUDNREF=N AND CODE 'Y' FOR THE PARTICULAR ENTITY TYPES
*      DESIRED.
*      USER DEFINED NESTS ARE NORMALLY DOCUMENTATIONAL ENTRIES WHICH
*      ARE NOT NEEDED FOR AN EXECUTABLE DIALOG OR APPLICATION.  SOME
*      USERS WISH TO ELIMINATE SUCH ENTRIES WHEN MIGRATING.  REVIEW THE
*      OPTIONS TAKEN FOR THE EXTRACTION OF USER DEFINED NESTS (XUDNXRT
*      AND ASSOCIATED PARAMETERS).  IF USER DEFINED NEST FOR AN ENTITY
*      TYPE ARE EXCLUDED FROM EXTRACTION, THEN REFER TO TECHNICAL
*      BULLETIN UM-9002-0003 FOR ADDITIONAL INFORMATION REGARDING THE
*      SUCCESSFUL MIGRATION OF USER DEFINED COMMENTS.
*-------------------------------------------------------------------
*
*  13. XUDNREL       (EXCLUDE USER DEFINED NEST REFERENCES FOR ELEMENTS)
*  -   PURPOSE: CREATE UPLOAD SYNTAX WITHOUT REFERENCES TO USER DEFINED
*      NESTS FOR ELEMENTS.
*  -   DEFAULT: THE SYNTAX IS CREATED INCLUDING REFERENCES TO USER
*      DEFINED NESTS FOR ELEMENTS
*  -   TO INVOKE THIS OPTION, CODE: XUDNREL=Y
*  -   TO USE THE DEFAULT, CODE: XUDNREL=N
*  -   COMMENTS: SEE COMMENTS UNDER XUDNREF.
*-------------------------------------------------------------------
*
*  14. XUDNRAT       (EXCLUDE USER DEFINED NEST REFERENCES FOR ATTRIBUTES)
*  -   PURPOSE: CREATE UPLOAD SYNTAX WITHOUT REFERENCES TO
*      USER DEFINED NESTS FOR ATTRIBUTES.
*  -   DEFAULT: THE SYNTAX IS CREATED INCLUDING REFERENCES TO USER
*      DEFINED NESTS FOR ATTRIBUTES
*  -   TO INVOKE THIS OPTION, CODE: XUDNRAT=Y
*  -   TO USE THE DEFAULT, CODE: XUDNRAT=N
*  -   COMMENTS: SEE COMMENTS UNDER XUDNREF.
*-------------------------------------------------------------------
```

```
*-------------------------------------------------------------------
*
*  15. XUDNRSY       (EXCLUDE USER DEFINED NEST REFERENCES FOR SYSTEMS)
*  -   PURPOSE: CREATE UPLOAD SYNTAX WITHOUT REFERENCES TO USER DEFINED
*      NESTS FOR SYSTEMS.
*  -   DEFAULT: THE SYNTAX IS CREATED INCLUDING REFERENCES TO USER
*      DEFINED NESTS FOR SYSTEMS
*  -   TO INVOKE THIS OPTION, CODE: XUDNRSY=Y
*  -   TO USE THE DEFAULT, CODE: XUDNRSY=N
*  -   COMMENTS: SEE COMMENTS UNDER XUDNREF.
*-------------------------------------------------------------------
*
*  16. XUDNRRC       (EXCLUDE USER DEFINED NEST REFERENCES FOR RECORDS)
*  -   PURPOSE: CREATE UPLOAD SYNTAX WITHOUT REFERENCES TO USER DEFINED
*      NESTS FOR RECORDS.
*  -   DEFAULT: THE SYNTAX IS CREATED INCLUDING REFERENCES TO USER
*      DEFINED NESTS FOR RECORDS
*  -   TO INVOKE THIS OPTION, CODE: XUDNRRC=Y
*  -   TO USE THE DEFAULT, CODE: XUDNRRC=N
*  -   COMMENTS: SEE COMMENTS UNDER XUDNREF.
*-------------------------------------------------------------------
*
*  17. XUDNRMD       (EXCLUDE USER DEFINED NEST REFERENCES FOR MODULES)
*  -   PURPOSE: CREATE UPLOAD SYNTAX WITHOUT REFERENCES TO USER DEFINED
*      NESTS FOR MODULES.
*  -   DEFAULT: THE SYNTAX IS CREATED INCLUDING REFERENCES TO USER
*      DEFINED NESTS FOR MODULES
*  -   TO INVOKE THIS OPTION, CODE: XUDNRMD=Y
*  -   TO USE THE DEFAULT, CODE: XUDNRMD=N
*  -   COMMENTS: SEE COMMENTS UNDER XUDNREF.
*-------------------------------------------------------------------
*
*  18. XUDNRPG       (EXCLUDE USER DEFINED NEST REFERENCES FOR PROGRAMS)
*  -   PURPOSE: CREATE UPLOAD SYNTAX WITHOUT REFERENCES TO USER DEFINED
*      NESTS FOR PROGRAMS.
*  -   DEFAULT: THE SYNTAX IS CREATED INCLUDING REFERENCES TO USER
*      DEFINED NESTS FOR PROGRAMS
*  -   TO INVOKE THIS OPTION, CODE: XUDNRPG=Y
*  -   TO USE THE DEFAULT, CODE: XUDNRPG=N
*  -   COMMENTS: SEE COMMENTS UNDER XUDNREF.
*-------------------------------------------------------------------
*
*  19. XUDNRUS       (EXCLUDE USER DEFINED NEST REFERENCES FOR USERS)
*  -   PURPOSE: CREATE UPLOAD SYNTAX WITHOUT REFERENCES TO USER DEFINED
*      NESTS FOR USERS.
*  -   DEFAULT: THE SYNTAX IS CREATED INCLUDING REFERENCES TO USER
*      DEFINED NESTS FOR USERS
*  -   TO INVOKE THIS OPTION, CODE: XUDNRUS=Y
*  -   TO USE THE DEFAULT, CODE: XUDNRUS=N
*  -   COMMENTS: SEE COMMENTS UNDER XUDNREF.
*-------------------------------------------------------------------
```

```
*-------------------------------------------------------------------
*
*  20. DBQUOTE       (DOUBLE QUOTE)
*  -   PURPOSE: USE A DOUBLE QUOTE (") THROUGHOUT IDD SYNTAX.
*  -   DEFAULT: A SINGLE QUOTE (') IS USED FOR ALL IDD SYNTAX.
*  -   TO INVOKE THIS OPTION, CODE: DBQUOTE=Y
*  -   TO USE THE DEFAULT, CODE: DBQUOTE=N
*  -   COMMENTS: THIS OPTION SHOULD BE USED AT SITES WHERE THE
*      DICTIONARY STANDARD IS A DOUBLE QUOTE.
*-------------------------------------------------------------------
*
*  21. EXNTWK        (EXTRACT IDMSNTWK)
*  -   PURPOSE: EXTRACT THE IDMSNTWK SCHEMA AND RELATED COMPONENT.
*  -   DEFAULT: NO PORTION OF THE IDMSNTWK SCHEMA IS MIGRATED.
*  -   TO INVOKE THIS OPTION, CODE: EXNTWK=Y
*  -   TO USE THE DEFAULT, CODE: EXNTWK=N
*  -   COMMENTS: THIS OPTION SHOULD ONLY BE USED IN VERY SPECIAL
*      CIRCUMSTANCES.  NORMALLY, EXTRACTION OF THE IDMSNTWK SCHEMA AND
*      ITS COMPONENTS WOULD CAUSE INCREASED PROCESSING TIME WITH NO
*      TANGIBLE RESULTS.  THE IDMSNTWK SCHEMA IS AVAILABLE TO EVERY
*      DICTIONARY.  MANY OF THE COMPONENTS CANNOT BE UPLOADED USING
*      IDMS UTILITIES.
*-------------------------------------------------------------------
*
*  22. XELEMNT       (EXCLUDE ELEMENTS)
*  -   PURPOSE: OMIT ALL ELEMENTS FROM MIGRATION.
*  -   DEFAULT: RELEVANT ELEMENTS ARE MIGRATED.
*  -   TO INVOKE THIS OPTION, CODE: XELEMNT=Y
*  -   TO USE THE DEFAULT, CODE: XELEMNT=N
*-------------------------------------------------------------------
*
*  23. XELECOB       (EXCLUDE ELEMENTS WHEN COBOLFORMAT IS USED)
*  -   PURPOSE: OMIT ALL ELEMENTS FROM MIGRATION WHEN COBOLFORMAT IS
*      USED FOR RECORDS.
*  -   DEFAULT: RELEVANT ELEMENTS ARE MIGRATED.
*  -   TO INVOKE THIS OPTION, CODE: XELECOB=Y
*  -   TO USE THE DEFAULT, CODE: XELECOB=N
*  -   COMMENTS: WHEN COBOLFORMAT IS USED, ELEMENTS REFERENCED IN THE
*      RECORDS ARE AUTOMATICALLY DEFINED WHEN THE RECORDS ARE ADDED.
*      THE MIGRATION OF ELEMENTS IS NOT NECESSARY. HOWEVER, IF
*      ADDITIONAL DOCUMENTATION HAS BEEN ADDED TO ELEMENTS, SUCH
*      DOCUMENTATION WOULD BE LOST UNLESS ELEMENTS ARE EXPLICITLY
*      MIGRATED.  IN SUCH CASES, THIS OPTION SHOULD NOT BE USED.
*-------------------------------------------------------------------
```

```
*-----------------------------------------------------------------------
*
*  24. EXSYREC       (EXTRACT SYSTEM RECORDS)
*  -   PURPOSE: EXTRACT CERTAIN SYSTEM RECORDS.
*  -   DEFAULT: THE RECORDS IN QUESTION ARE OMITTED FROM MIGRATION.
*  -   TO INVOKE THIS OPTION, CODE: EXSYREC=Y
*  -   TO USE THE DEFAULT, CODE: EXSYREC=N
*  -   SYSTEM RECORDS ARE:
*           ADSO-APPLICATION-GLOBAL-RECORD
*           ADSO-APPLICATION-MENU-RECORD
*           ADSO-STAT-DEF-REC
*           SUBSCHEMA-CTRL
*  -   COMMENTS: THE SYSTEM RECORDS ARE NORMALLY OMITTED FROM
*      MIGRATION. THESE RECORDS ARE IN EVERY DICTIONARY AND USUALLY
*      HAVE NO CHANGES.  BECAUSE MIGRATION WITHOUT CHANGEONLY WOULD
*      CREATE DELETE RECORD SYNTAX FOR THESE RECORDS AND THUS
*      DISCONNECT THE RECORDS FROM ALL DIALOGS CURRENTLY USING THEM IN
*      THE TARGET DICTIONARY, THEY SHOULD BE OMITTED FROM MIGRATION.
*      THIS OPTION SHOULD ONLY BE USED FOR SPECIAL MIGRATIONS WHEN ONE
*      OF THESE RECORDS HAS CHANGED AND PROCESSING IS PLANNED TO
*      REGENERATE ALL AFFECTED DIALOGS IN THE TARGET DICTIONARY.
*-----------------------------------------------------------------------
*
*      XUDNXRT   (EXCLUDE ALL USER DEFINED NEST FROM EXTRACTION)
*  -   PURPOSE: OMIT ENTITIES RELATED BY USER DEFINED NESTS.
*  -   DEFAULT: ENTITIES RELATED BY USER DEFINED NESTS ARE EXTRACTED.
*  -   TO INVOKE THIS OPTION, CODE: XUDNXRT=Y
*  -   TO USE THE DEFAULT, CODE: XUDNXRT=N
*  -   COMMENTS: IF THIS OPTION IS 'Y' ALL OF THE FLAGS BEGINING XUDNX
*      ARE SET TO 'Y'.  IT IS NOT POSSIBLE TO OVERRIDE THIS OPTON ON AN
*      INDIVIDUAL ENTITY BASIS.  IF USER DEFINED NESTS EXTRACTION IS
*      DESIRED FOR SOME ENTITY TYPES, BUT NOT OTHERS, CODE XUDNXRT=N
*      AND CODE 'Y' FOR THE PARTICULAR ENTITY TYPES DESIRED.
*
*      DURING THE EXTRACTION PHASE OF MIGRATION, DICTIONARY MIGRATOR
*      FOLLOWS USER DEFINED NESTS AS WELL AS SYSTEM DEFINED NESTS TO
*      FIND ALL ENTITIES RELATED TO THE ENTITY NAMED ON THE EXTRACT
*      STATEMENT.  IN SOME CASES, THE RELATIONSHIPS FOUND ARE TENUOUS
*      OR DOCUMENTATION NOT RELATED TO THE PURPOSE OF THE MIGRATION.
*      IN SUCH CASES, USER DEFINED NESTS MAY BE EXCLUDED FROM
*      EXTRACTION IN ORDER TO LIMIT THE NUMBER OF ENTITIES MIGRATED.
*      FURTHERMORE, OCCASIONALLY A SYSTEM NEST AND A USER DEFINED NEST
*      MAY RELATE THE SAME ENTITIES BY DIFFERENT PATHS.  DICTIONARY
*      MIGRATOR MAY THEN PRODUCE A MESSAGE 'ES00514E - ENTIY NEST
*      EXPLOSION TABLE SIZE EXCEEDED'.  THESE OPTIONS MAY BE USED TO
*      CIRCUMVENT THIS CONDITION.
*-----------------------------------------------------------------------
```

```
*-------------------------------------------------------------------
*
*  25. XUDNXEL       (EXCLUDE EXTRACTION OF USER DEFINED NEST - ELEMENTS)
*  -   PURPOSE: OMIT ELEMENTS RELATED BY USER DEFINED NESTS.
*  -   DEFAULT: ELEMENTS RELATED BY USER DEFINED NESTS ARE EXTRACTED.
*  -   TO INVOKE THIS OPTION, CODE: XUDNXEL=Y
*  -   TO USE THE DEFAULT, CODE: XUDNXEL=N
*  -   COMMENTS: SEE COMMENTS UNDER XUDNXRT.
*-------------------------------------------------------------------
*
*  26. XUDNXRC
*  -   PURPOSE: RESERVED BYTE
*  -   CODE:        XUDNXRC=N
*  -   COMMENTS: THIS OPTION IS RESERVED FOR FUTURE USE.
*-------------------------------------------------------------------
*
*  27. XUDNXMD       (EXCLUDE EXTRACTION OF USER DEFINED NEST - MODULES)
*  -   PURPOSE: OMIT MODULES RELATED BY USER DEFINED NESTS.
*  -   DEFAULT: MODULES RELATED BY USER DEFINED NESTS ARE EXTRACTED.
*  -   TO INVOKE THIS OPTION, CODE: XUDNXMD=Y
*  -   TO USE THE DEFAULT, CODE: XUDNXMD=N
*  -   COMMENTS: SEE COMMENTS UNDER XUDNXRT.
*-------------------------------------------------------------------
*
*  28. XUDNXUS       (EXCLUDE EXTRACTION OF USER DEFINED NEST - USERS)
*  -   PURPOSE: OMIT USERS RELATED BY USER DEFINED NESTS.
*  -   DEFAULT: USERS RELATED BY USER DEFINED NESTS ARE EXTRACTED.
*  -   TO INVOKE THIS OPTION, CODE: XUDNXUS=Y
*  -   TO USE THE DEFAULT, CODE: XUDNXUS=N
*  -   COMMENTS: SEE COMMENTS UNDER XUDNXRT.
*-------------------------------------------------------------------
*
*  29. XUDNXAT      (EXCLUDE EXTRACTION OF USER DEFINED NEST - ATTRIBUTES)
*  -   PURPOSE: OMIT ATTRIBUTES RELATED BY USER DEFINED NESTS.
*  -   DEFAULT: ATTRIBUTES RELATED BY USER DEFINED NESTS ARE EXTRACTED.
*  -   TO INVOKE THIS OPTION, CODE: XUDNXAT=Y
*  -   TO USE THE DEFAULT, CODE: XUDNXAT=N
*  -   COMMENTS: SEE COMMENTS UNDER XUDNXRT.
*-------------------------------------------------------------------
*
*  30. XUDNXSY       (EXCLUDE EXTRACTION OF USER DEFINED NEST - SYSTEMS)
*  -   PURPOSE: OMIT SYSTEMS RELATED BY USER DEFINED NESTS.
*  -   DEFAULT: SYSTEMS RELATED BY USER DEFINED NESTS ARE EXTRACTED.
*  -   TO INVOKE THIS OPTION, CODE: XUDNXSY=Y
*  -   TO USE THE DEFAULT, CODE: XUDNXSY=N
*  -   COMMENTS: SEE COMMENTS UNDER XUDNXRT.
*-------------------------------------------------------------------
```

```
*-------------------------------------------------------------------
*
*  31. XIMSYNR       (SUPPRESS THE SYNTAX FILE DISPLAY REPORT)
*  -   PURPOSE: SUPPRESS THE SYNTAX FILE DISPLAY REPORT WHEN RUN=IMPORT
*  -   DEFAULT: THE SYNTAX FILE DISPLAY REPORT IS PRODUCED WHEN
*      RUN=IMPORT OR RUN=AUDIT.
*  -   TO INVOKE THIS OPTION, CODE: XIMSYNR=Y
*  -   TO USE THE DEFAULT, CODE: XIMSYNR=N
*  -   COMMENTS: THE SYNTAX FILE DISPLAY REPORT PRINTS THE CONTENTS OF
*      ALL SYNTAX FILES.  WHEN RUN=IMPORT, AND PARTICULARLY WITH
*      CHANGEONLY THIS REPORT IS USEFUL, BUT NOT ESSENTIAL.  IF THE
*      USER BELIEVES THAT THE REPORT IS NOT NEEDED, THIS OPTION MAY BE
*      USED. THE OPTION DOES NOT APPLY TO RUN=AUDIT AS THE ONLY
*      DIFFERENCE BETWEEN RUN=MIGRATE AND RUN=AUDIT IS THE CREATION OF
*      THIS REPORT.
*-------------------------------------------------------------------
*
*  32. DELADDS       (USE DELETE AND ADD VERBS FOR SYNTAX)
*  -   PURPOSE: INSTEAD OF MODIFYING ENTITIES IN THE OBJECT DICTIONARY,
*      DELETE THE ENTITY AND ADD IT LATER.
*  -   DEFAULT: ENITITIES WILL BE MODIFIED WHENEVER POSSIBLE.
*  -   TO INVOKE THIS OPTION, CODE: DELADDS=Y
*  -   TO USE THE DEFAULT, CODE: DELADDS=N
*  -   COMMENTS: THE MODIFY VERB INSURES THAT EXISTING RELATIONSHIPS IN
*      THE OBJECT DICTIONARY WILL NOT BE LOST WHEN A MIGRATION UPDATES
*      AN ENTITY.  IN SOME CIRCUMSTANCES, A USER MAY WISH TO STILL USE
*      DELETE/ADD.  IF THIS OPTION IS USED, SYNTAX IS CREATED WITHOUT
*      ACCESSING THE TARGET DICTIONARY;  ALSO, THE DDDLDEL FILE
*      CONTAINS VALID SYNTAX AND SHOULD BE PART OF THE UPLOAD PROCESS.
*      NOTE: THIS OPTION INVOKES PROCESSING THAT IS THE SAME AS
*      NON-CHANGEONLY PROCESSING PRIOR TO RELEASE 12.0.
*-------------------------------------------------------------------
*
*  33. EXTSAME       (EXTRACT SAME AS RELATIONSHIPS)
*  -   PURPOSE: INCLUDE IN THE EXTRACTION PHASE ENTITIES WHICH ARE
*      RELATED TO EXTRACTED ENTITIES BY A SAME AS RELATIONSHIP.
*  -   DEFAULT: SAME AS RELATIONSHIPS ARE IGNORED.
*  -   TO INVOKE THIS OPTION, CODE: EXTSAME=Y
*  -   TO USE THE DEFAULT, CODE: EXTSAME=N
*  -   COMMENTS: USING THE DEFAULT VALUE LIMITS THE SCOPE OF THE
*      MIGRATION AND ALSO CAN AVOID TABLE OVERFLOW DUE TO MULTIPLE
*      RELATIONSHIPS BETWEEN TWO ENTITIES.
*-------------------------------------------------------------------
*
*  34. DBABEND       (ABEND ON DATABASE ERROR)
*  -   PURPOSE: FORCE AN ABEND (AND A DUMP) IF AN UNEXPECTED ERROR
*      STATUS IS RETURNED FROM A DATABASE CALL.
*  -   DEFAULT: UNEXPECTED ERROR STATUS RESULT IN PROGRAM TERMINATION
*      WITH USER CONDITION CODE 2222.
*  -   TO INVOKE THIS OPTION, CODE: DBABEND=Y
*  -   TO USE THE DEFAULT, CODE: DBABEND=N
*  -   COMMENTS: IF THIS OPTION IS TAKEN, THE SYSTEM COMPLETION CODE
*      WILL BE S0C1.
*      REGARDLESS OF THE VALUE OF THIS OPTION, THE RELEVANT CONTENTS OF
*      SUBSCHEMA-CTRL WILL BE DISPLAYED IN THE AUDIT FILE.
*      IN MOST CASES, THIS INFORMATION IS SUFFICIENT FOR PROBLEM
*      DETERMINATION.
*      THIS OPTION IS ONLY AVAILABLE FOR OS AND VM.  IN A DOS
*      ENVIRONMENT, A UNEXPECTED DATABASE ERROR STATUS WILL ALWAYS
*      PRODUCE AN OPERATION EXCEPTION.
*-------------------------------------------------------------------
```

```
*
*  35. NOEXATT       (OMIT EXTRACTION OF CLASS-ATTRIBUTES)
*  -   PURPOSE: MIGRATE ENTITIES INCLUDING ANY REFERENCE TO ATTRIBUTES
*      BUT DO NOT MIGRATE ANY CLASS-ATTRIBUTE STRUCTURES.
*  -   DEFAULT: ALL ENTITY TYPES ARE MIGRATED
*  -   TO INVOKE THIS OPTION, CODE: EXNOATT=Y
*  -   TO USE THE DEFAULT, CODE: EXNOATT=N
*  -   COMMENTS: THIS OPTION SHOULD ONLY BE CONSIDERED WHEN THE
*      CHANGEONLY PARAMETER CANNOT BE USED.  THE EXTRACTION OF
*      CLASS-ATTRIBUTE STRUCTURES MAY SIGNIFICANTLY LENGTHEN RUN TIMES
*      OF MIGRATIONS.  AS ATTRIBUTES ARE A DOCUMENTATIONAL ENTITIES,
*      THEY TEND TO HAVE A LOW VOLATILITY.  HENCE, IT IS NOT NECESSARY
*      TO MIGRATE THEM ON EVERY MIGRATION.  THIS OPTION ELIMINATES THE
*      EXTRACTION OF ATTRIBUTES, BUT ALL REFERENCES TO THE ATTRIBUTES
*      ARE RETAINED IN ALL OTHER ENTITY OCCURRENCES.  IF THE TARGET
*      DICTIONARY CONTAINS THE SAME CLASS-ATTRIBUTE STRUCTURES, ALL
*      DOCUMENTATION WILL BE PRESERVED.
*      WHEN USING THIS OPTION, THE USER MUST INSURE THAT ALL CLASSES
*      TO BE REFERENCED EXIST IN THE TARGET DICTIONARY AND THAT ALL
*      ATTRIBUTES TO BE REFERENCED EITHER EXIST OR WILL BE ADDED
*      AUTOMATICALLY TO THE TARGET DICTIONARY.
*      FOR RELEASE 3.5 USERS, TECHNICAL BULLETIN UM-9012-0014 MUST BE
*      APPLIED.
*-----------------------------------------------------------------------
```

```
*-----------------------------------------------------------------------
*
*  36. NOEXCLS       (OMIT EXTRACTION OF CLASS)
*  -   PURPOSE: MIGRATE ATTRIBUTES BUT DO NOT MIGRATE CLASSES.
*      (THE CLASS WILL STILL BE REFERENCED IN THE ATTRIBUTE STATEMENT.)
*  -   DEFAULT: ALL ENTITY TYPES ARE MIGRATED
*  -   TO INVOKE THIS OPTION, CODE: EXNOCLS=Y
*  -   TO USE THE DEFAULT, CODE: EXNOCLS=N
*  -   COMMENTS: THIS OPTION SHOULD ONLY BE CONSIDERED WHEN THE
*      CHANGEONLY PARAMETER CANNOT BE USED.
*      AS CLASS ENTITIES ARE CHANGED INFREQUENTLY, IT MAY NOT BE
*      NECESSARY TO INCLUDE THEM IN EVERY MIGRATION.   USING THIS
*      OPTION WILL CAUSE EXTRACTION OF ATTRIBUTE STRUCTURES, BUT NO
*      CLASSES WILL BE MIGRATED.
*      WHEN USING THIS OPTION, THE USER MUST INSURE THAT ALL CLASSES
*      TO BE REFERENCED EXIST IN THE TARGET DICTIONARY.
*      FOR RELEASE 3.5 USERS, TECHNICAL BULLETIN UM-9012-0014 MUST BE
*      APPLIED.
*-----------------------------------------------------------------------
*
*  37. NOEXSYS       (OMIT EXTRACTION OF SYSTEMS)
*  -   PURPOSE: MIGRATE ENTITIES INCLUDING REFERENCES TO SYSTEMS
*      BUT DO NOT MIGRATE SYSTEM ENTITY OCCURRENCES.
*  -   DEFAULT: ALL ENTITY TYPES ARE MIGRATED
*  -   TO INVOKE THIS OPTION, CODE: EXNOSYS=Y
*  -   TO USE THE DEFAULT, CODE: EXNOSYS=N
*  -   COMMENTS: THIS OPTION SHOULD ONLY BE CONSIDERED WHEN THE
*      CHANGEONLY PARAMETER CANNOT BE USED.
*      AS SYSTEM ENTITIES ARE CHANGED INFREQUENTLY, IT MAY NOT BE
*      NECESSARY TO INCLUDE THEM IN EVERY MIGRATION.   USING THIS
*      OPTION WILL ALL REFERENCES TO SYSTEMS TO BE PRESERVED IN ANY
*      ENTITY OCCURRENCE, BUT NO SYSTEM ENTITIES OCCURRENCES WILL BE
*      MIGRATED.
*      WHEN USING THIS OPTION, THE USER MUST INSURE THAT ALL SYSTEMS
*      TO BE REFERENCED EXIST IN THE TARGET DICTIONARY.
*      FOR RELEASE 3.5 USERS, TECHNICAL BULLETIN UM-9012-0014 MUST BE
*      APPLIED.
*-----------------------------------------------------------------------
```

```
*-------------------------------------------------------------------
*
*  38. STOPVER       (STOP AFTER VALIDATION ERROR)
*  -   PURPOSE: WHEN A CRITICAL LEVEL ERROR IS ENCOUNTERED DURING
*      VALIDATION, STOP EXECUTION AT THE END OF THE VALIDATION PROCESS.
*  -   DEFAULT: EXECUTION CONTINUES UNTIL ALL PROCESSING DEFINED BY THE
*      RUN TYPE IS COMPLETED.
*  -   TO INVOKE THIS OPTION, CODE: STOPVER=Y
*  -   TO USE THE DEFAULT, CODE: STOPVER=N
*  -   COMMENTS: THIS OPTION IS ONLY RELEVANT WHEN THE RUN TYPE IS
*      MIGRATE, AUDIT, OR IMPORT.
*      IF THIS OPTION IS USED, A CRITICAL ERROR WILL TERMINATE
*      PROCESSING BEFORE THE SYNTAX IS CREATED, AND A CONDITION CODE OF
*      8 WILL ALSO BE SET IN MVS.  IF NO ERRORS ARE DETECTED DURING
*      VALIDATION, THE SYNTAX WILL BE CREATED.  THE SAME JOB CAN
*      CONTAIN THE UPLOAD STEPS USING CONDITIONAL PROCESSING WHICH
*      CHECKS THE CC OF THE MIGRATOR STEP. THE UPLOAD STEPS WOULD ONLY
*      BE RUN WHEN THERE ARE NO ERRORS REQUIRING REVIEW.
*      NOTE: NO MESSAGES HAVE A DEFAULT SEVERITY OF CRITICAL.  THE USER
*      MUST DECIDE WHICH ERRORS SHOULD BE CONSIDERED CRITICAL AND
*      UPDATE THE MESSAGE SEVERITY TABLE ACCORDINGLY.
*-------------------------------------------------------------------
*
*  39. NOATRXP       (DO NOT EXPLODE ATTRIBUTE NETWORK IF LEVEL=ONLY)
*  -   PURPOSE: IF LEVEL=ONLY MIGRATION IS SPECIFIED FOR CLASS, CLSATTR
*      OR ATTRIBUTES THE ATTRIBUTE EXPLOSION SET IS FOLLOWED WHICH CAN
*      RESULT IN THE MIGRATION OF A NETWORK OF ATTRIBUTES.
*  -   DEFAULT: EXPLOSION SETS ARE FOLLOWED
*  -   TO INVOKE THIS OPTION, CODE: NOATRXP=Y
*  -   TO USE THE DEFAULT, CODE: NOATRXP=N
*  -   COMMENTS: THIS OPTION ONLY APPLIES IN THE CASE OF A LEVEL=ONLY
*      MIGRATION.
*      IF THIS OPTION IS APPLIED THEN ONLY THE REFERENCED ATTRIBUTE
*      (ATTRIBUTE MIGRATION) OR ATTRIBUTES WITHIN THE CLASS (CLASS
*      MIGRATION) WILL BE EXTRACTED FROM THE SOURCE DICTIONARY.
*-------------------------------------------------------------------
*
*  40. NOSAUTH       (BYPASS SOURCE DICTIONARY SECURITY CHECKING)
*  -   PURPOSE: USERID/PASSWORD ARE VERIFIED AS HAVING DISPLAY
*      AUTHORITY IN THE SOURCE DICTIONARY, FOR SIGNON, OVERRIDE
*      AUTHORIZATION (IF SPECIFIED) AND FOR EACH EXTRACTED ENTITY TYPE.
*      IF THIS AUTHORIZATION CHECK FAILS MIGRATOR WILL ABORT.
*  -   DEFAULT: SECURITY CHECKING WILL BE PERFORMED
*  -   TO INVOKE THIS OPTION, CODE: NOSAUTH=Y
*  -   TO USE THE DEFAULT, CODE: NOSAUTH=N
*-------------------------------------------------------------------
```

```
*-----------------------------------------------------------------------
*
*  41. NOTAUTH       (BYPASS TARGET DICTIONARY SECURITY CHECKING)
*  -   PURPOSE: USERID/PASSWORD ARE VERIFIED AS HAVING UPDATE AUTHORITY
*      IN THE TARGET DICTIONARY, FOR SIGNON, OVERRIDE AUTHORIZATION (IF
*      SPECIFIED) AND FOR EACH EXTRACTED ENTITY TYPE. IF THIS
*      AUTHORIZATION CHECK FAILS MIGRATOR WILL ABORT.
*  -   DEFAULT: SECURITY CHECKING WILL BE PERFORMED
*  -   TO INVOKE THIS OPTION, CODE: NOTAUTH=Y
*  -   TO USE THE DEFAULT, CODE: NOTAUTH=N
*-----------------------------------------------------------------------
*
*-----------------------------------------------------------------------
*
*  43. ABGNSRC       (ADSOBGEN SOURCE FOR DIALOGS)
*  -   PURPOSE: FORMAT THE ADSOBGN FILE SO IT MAY BE USED AS INPUT TO A
*      CULPRIT REPORT WHICH CREATES SYNTAX FOR 'GENERATE FROM SOURCE'
*      FOR THE ADSOBCOM UTILITY.
*  -   DEFAULT: 'GENERATE FROM LOAD ' SYNTAX IS CREATED.
*  -   TO INVOKE THIS OPTION, CODE: ABGNSRC=Y
*  -   TO USE THE DEFAULT, CODE: ABGNSRC=N
*  -   COMMENTS: THIS OPTION IS PRIMARILY FOR CASES WHERE ADSOBGEN
*      SOURCE STATEMENT ARE USEFUL.
*-----------------------------------------------------------------------
*
*  44. XSIGNON       (OMIT SIGNON FROM SYNTAX FILES)
*  -   PURPOSE: FORMAT ALL SYNTAX FILES WITHOUT A SIGNON STATEMENT.
*  -   DEFAULT: SIGNON STATEMENTS USING DATA FROM RUN TIME PARAMETERS
*               IS GENERATED FOR AS APPROPRIATE FOR EACH SYNTAX FILE.
*  -   TO INVOKE THIS OPTION, CODE: XSIGNON=Y
*  -   TO USE THE DEFAULT, CODE: XSIGNON=N
*  -   COMMENTS: THIS OPTION ALLOWS A SEPARATE FILE WITH SIGNON
*      INFORMATION TO BE CONCATENATED TO THE SYNTAX FILE AT UPLOAD
*      TIME.
*      WARNING: CERTAIN FILES MAY BE EMPTY IF THIS OPTION IS USED WHEN
*      NO SEPARATE SIGNON STATEMENT FILE IS CONCATENATED AND NO
*      OCCURRENCES OF A GIVEN ENTITY TYPE ARE MIGRATED.  EMPTY FILES
*      WILL CAUSE THE UPLOAD UTILITIES TO ABEND.
*
*-----------------------------------------------------------------------
```

```
*------------------------------------------------------------------------
*
*  45. XSIGMAP       (OMIT SIGNON FROM MAP SYNTAX FILES RHDCDEL/RHDCUPD
*  -   PURPOSE: FORMAT MAP COMPILER SYNTAX FILES WITHOUT A SIGNON.
*      DEFAULT: SIGNON STATEMENTS USING DATA FROM RUN TIME PARAMETERS
*               IS GENERATED FOR AS APPROPRIATE FOR MAP SYNTAX FILE.
*  -   TO INVOKE THIS OPTION, CODE:        XSIGMAP=N
*  -   COMMENTS: THE BATCH MAPPING COMPILER RHDCMAP1 HAS BEEN ENHANCED
*      TO ALLOW ACCESS TO A SIGNON-REQUIRED DICTIONARY WHEN NO SIGNON
*      CARD IS PROVIDED IF THE USER-ID OF THE PERSON WHO SUBMITTED THE
*      JOB HAS ACCESS TO THE DICTIONARY. THIS ALLOWS DICTIONARY
*      MIGRATOR USERS TO UTILIZE THIS FACILITY REGARDLESS OF THE FORMAT
*      OF THE OTHER SYNTAX FILES.
*      WARNING: MAP SYNTAX FILES MAY BE EMPTY IF THIS OPTION IS USED
*      WHEN NO SEPARATE SIGNON STATEMENT FILE IS CONCATENATED AND NO
*      OCCURRENCES OF A GIVEN ENTITY TYPE ARE MIGRATED.  EMPTY FILES
*      WILL CAUSE THE UPLOAD UTILITIES TO ABEND.
*------------------------------------------------------------------------
*
*  46. XEQUDAT      (SKIP EXTRACTION OF ENTITIES WITH EQUAL DATES)
*  -   PURPOSE: TO AVOID EXTRACTION OF ENTITIES WITH EQUAL DATES WHERE
*               TIMESTAMPS ARE NOT SUPPORTED IN CHANGEONLY MIGRATION.
*  -   DEFAULT: ENTITIES WITH EQUAL DATES AND NO TIME STAMPS WILL BE
*               MARKED FOR EXTRACTION IN A CHANGEONLY MIGRATION.
*  -   TO INVOKE THIS OPTION, CODE:        XEQUDAT=Y
*  -   TO USE THE DEFAULT, CODE; XEQUDAT=N
*  -   COMMENTS: DATE AND TIMESTAMPS ARE USED ARE THE BASIS FOR
*      COMPARISON IN A CHANGEONLY MIGRATION. WHERE DATES ARE EQUAL AND
*      NO TIMESTAMP IS SUPPORTED, THE ENTITY WILL BE MARKED FOR
*      EXTRACTION. THIS CAN RESULT IN UNNECESSARY MIGRATION OF MANY
*      ENTITIES. ELEMENTS ARE AN EXAMPLE. BY SETTING XEQUDAT=Y YOU WILL
*      AVOID THE MIGRATION OF SUCH ENTITIES.
*------------------------------------------------------------------------
```

G.4 CA-IDMS/Dictionary Migrator Assistant Runtime Parameters

```
*--------------------------------------------------------------------
* DICTIONARY MIGRATOR ASSISTANT RUNTIME PARAMETERS
*--------------------------------------------------------------------
*
*XDMCPARM THIS IS THE INSTALLATION TAILORING MACRO USED BY THE DMA
*SYSTEM TO PROVIDE RUNTIME VALUES.
*OPERANDS:
*                 DMATSK='1-8 CHAR'    TASK USED TO INVOKE DMA.
*                 HLPDICT='1-8 CHAR'   ALTERNATE DICTIONARY USED FOR
*                                      GSIHELP.
*                 HLPNODE='1-8 CHAR'   ALTERNATE NODE USED FOR GSIHELP.
*
*                 HLPVERS=INTEGER      VERSION NUMBER OF HELP MODULES.
*
*ASSEMBLED VALUES AT INSTALLATION:
*       XDMCPARM DMATSK='DMA    ',
*                HLPDICT='        ',
*                HLPNODE='        ',
*                HLPVERS=0
*
*--------------------------------------------------------------------
```

G.5 CA-IDMS/Dictionary Module Editor Runtime Parameters

```
 *--------------------------------------------------------------------
 * CA-IDMS/DICTIONARY MODULE EDITOR RUNTIME PARAMETERS
 *--------------------------------------------------------------------
         PRINT OFF
         COPY USECPARM
         PRINT ON
USETPARM CSECT                      CONTROL TABLE FOR DME
 *--------------------------------------------------------------------
 *   LEAVE IN UPPER CASE, MACRO PARAMETERS ARE CASE SENSITIVE MUST USE
 *                        UPPER CASE.
 *--------------------------------------------------------------------
 *            MODIFY PRODUCT TUNING PARAMETERS
 *            -------------------------------
 *USETPARM -- THIS MEMBER IS USED TO SPECIFY THE RUN-TIME VALUES TO BE
 *            USED AS INPUT TO THE INSTALLATION TAILORING MACRO,
 *            USECPARM, WHICH IS DYNAMICALLY LOADED BY ONLINE MENU
 *            USEAMEN, AND THE ACTIVE AND PASSIVE D.M.E. MODULES.
 *
 *            RUN-TIME VARIABLES
 *            ------------------
 *
 *            HLPDICT=(1-8 CHAR)   ALTERNATE DICT FOR ONLINE HELP
 *            HLPNODE=(1-8 CHAR)   ALTERNATE NODE FOR ONLINE HELP
 *            HLPVERS=(1-9999 NUM) VERSION NUMBER OF HELP MODULES.
 *
 *            LOCK=(Y/N)        IDD DB LOCKING  (YES OR NO)
 *                              Y = LONGTERM DBKEY LOCKS ARE SET ON A
 *                                  MODULE WHEN AN EDIT SESSION IS
 *                                  STARTED
 *
 *                              N = LONGTERM DBKEY LOCKS ARE NOT SET,
 *                                  SHOULD ONLY BE DONE ON ADVICE FROM
 *                                  CA TECHNICAL STAFF.
 *
 *            SCROLL=PAGE     SCROLL AMOUNT                          *
 *                                PAGE|HALF|CSR                      *
 *                                                                   *
 *            DELIMIT=;       COMMAND DELIMIT
 *
 *            PAD=            PAD  CHARACTER
 *                                 N|B     NULLS|BLANKS
 *
 *            VERSION=HIGHEST  DEFAULT IDD VERSION NUMBER
 *                                 HIGHEST - SELECT THE HIGHEST VER
 *                                 LOWEST  - SELECT THE LOWEST  VER
 *
 *            SECURTY=I         SECURTY SYSTEM IN FORCE (RESERVED)
 *                                 I = IDD (DEFAULT)
 *                                 D = DBMS
 *                                 B = DBMS AND IDD
 *
```

```
*            USERID =INPUT    ALLOW CHANGES TO USERID FROM WITHIN DME
*                             SESSION
*                                 INPUT = USERID CHANGE ALLOWED
*                                 PROT  = USERID CHANGE NOT ALLOWED
*
*            MODSORT=Y        DEFAULT TO SORTED MODULE LIST
*                                 Y = MODULE SORT ASSUMED
*                                 N = MODULE SORT NOT ASSUMED (LARGE
*                                       SHOP OPTION)
*
*            SETDB=N         RESET DATABASE
*                                 Y = RESET DATABASE/NODE TO ORIGINAL
*                                     VALUE ON DME ENTRY
*                                 N = DO NOT RESET DATABASE/NODE DEFAULT
*
*            CLRKEND=Y        CLEAR KEY = END
*                                 Y = CLEAR KEY = END ORIGINAL VALUE
*                                     ON DME ENTRY DEFAULT
*                                 N = CLEAR KEY = RESHOW
*
*  DEFAULT VALUES AS SUPPLIED WITH INSTALLATION:                    *
*       USECPARM HLPDICT='        ',   HELP DICTIONARY              *
*                HLPNODE='        ',   HELP NODE                    *
*                HLPVERS=1,            VERSION OF HELP TEXT         *
*                   LOCK=Y,            LOCK (YES|NO)                *
*                 SCROLL=PAGE,         SCROLL AMOUNT                *
*                DELIMIT=;,            COMMAND DELIMIT              *
*                    PAD=N,            PAD  CHARACTER               *
*                VERSION=HIGHEST,      DEFAULT IDD VERSION NUMBER   *
*               SECURTY=I,             SECURTY SYSTEM IN FORCE      *
*                 USERID=INPUT,        CHANGE ALLOWED TO USER ID    *
*                MODSORT=Y,            MODULE SORT ON               *
*                  SETDB=Y,            SET DATABASE                 *
*                CLRKEND=Y             CLEAR KEY IS END  COMMAND    *
*-------------------------------------------------------------------*
        USECPARM HLPDICT=TOOLDICT,                                   X
              HLPNODE=,                                              X
              HLPVERS=1,                                             X
              LOCK=Y,                                                X
              SCROLL=PAGE,                                           X
              DELIMIT=;,                                             X
              PAD=N,                                                 X
              VERSION=HIGHEST,                                       X
              SECURTY=I,                                             X
              USERID=INPUT,                                          X
              MODSORT=Y,                                             X
              SETDB=Y,                                               X
              CLRKEND=Y
        END
*-------------------------------------------------------------------
```

G.6 CA-IDMS/Dictionary Query Facility Runtime Parameters

```
*--------------------------------------------------------------------
* CA-IDMS/DICTIONARY QUERY FACILITY RUNTIME PARAMETERS
*--------------------------------------------------------------------
*
*DADTPARM THIS IS THE INSTALLATION TAILORING MACRO USED BY THE DQF
*SYSTEM TO PROVIDE RUNTIME VALUES.
*OPERANDS:
*                HLPDICT='1-8 CHAR'   ALTERNATE DICTIONARY USED FOR
*                                     GSIHELP.
*                HLPNODE='1-8 CHAR'   ALTERNATE NODE USED FOR GSIHELP.
*
*                HLPVERS=INTEGER      VERSION NUMBER OF HELP MODULES.
*
*ASSEMBLED VALUES AT INSTALLATION:
*       DADCPARM HLPDICT='        ',
*                HLPNODE='        ',
*                HLPVERS=1
*       END
*--------------------------------------------------------------------
```

G.7 CA-IDMS/DML Online Runtime Parameters

```
 *--------------------------------------------------------------------
 * CA-IDMS/DML-ONLINE RUNTIME PARAMETERS                              *
 *--------------------------------------------------------------------
 *                                                                    *
        PRINT OFF                                                    *
        COPY  USDCPARM                                               *
        PRINT ON                                                     *
USDTPARM CSECT                        CONTROL TABLE FOR DMLO         *
 *------------------------------------------------------------------ *
 * USDTPARM  --  IS THE INSTALLATION TAILORING MODULE USED BY DML/O TO *
 *               PROVIDE CUSTOM RUNTIME AND DEFAULT VALUES            *
 *                                                                    *
 * USDTPARM IS AN INDEPENDENT LOAD MODULE WHICH INCORPORATES VALUES    *
 * GENERATED BY USDCPARM, AS WELL AS VARIOUS TABLES.                  *
 *                                                                    *
 * IT IS LOADED AT RUN TIME BY PROGRAM USDTPIFN.                      *
 *                                                                    *
 * 14.0.1  01/08/99  DEVDE01  ADD SUPPORT FOR DEFENTK PARAMETER (38)  *
 *                                                                    *
 *------------------------------------------------------------------ *
 *                                                                    *
 *       YOUR RESPONSIBILITY AS INSTALLER IS TO :                     *
 *                                                                    *
 *   1.  SELECT APPROPRIATE VALUES FOR THE MACRO PARAMETERS AT THE END *
 *                                                                    *
 *   2.  UPDATE THE FOLLOWING SOURCE MODULES AS APPROPRIATE :         *
 *                                                                    *
 *       .. USD@MOPS            MENU-MODE DML OP CODES                *
 *       .. USD@MTXT            MENU-MODE DESCRIPTIVE TEXT            *
 *       .. USD@MSTL            MENU-MODE STATIC AREA DESCRIPTION     *
 *       .. USD@SSEX            SUBSCHEMA EXCLUSION LIST              *
 *       .. USD@DSPC            DISPLAYABLE CHARACTERS                *
 *       .. USD@KYWD            STANDARD ABBREVIATIONS                *
 *   3.  ASSEMBLE AND LINKEDIT PROGRAM USDTPARM                       *
 *------------------------------------------------------------------**
 * FOLLOWING IS AN EXPLANATION FOR EACH PARAMETER OF MACRO USDCPARM   *
 *                                                                    *
 *------------------------------------------------------------------**
 *                                                                    *
 * (1)  : ==>   HLPDICT    ALTERNATE DICTIONARY USED FOR HELP MODULES  *
 *                                                                    *
 *         DICTIONARY NAME (DICTNAME) OF DICTIONARY INTO WHICH THE ONLINE
 *         DOCUMENTATION / HELP MODULES HAVE BEEN PLACED PARAMETER IS  *
 *         OPTIONAL, DEFAULT VALUE IS '        '.                     *
 *------------------------------------------------------------------**
 *                                                                    *
 * (2)  : ==>   HLPNODE    ALTERNATE DICTNODE   USED FOR HELP MODULES  *
 *                                                                    *
 *         DICTIONARY NODE (DICTNODE) OF DICTIONARY INTO WHICH THE ONLINE
 *         DOCUMENTATION / HELP MODULES HAVE BEEN PLACED PARAMETER IS  *
 *         OPTIONAL, DEFAULT VALUE IS '        '.                     *
 *------------------------------------------------------------------**
```

```
*                                                                     *
* (3)  : ==>   HLPVERS    VERSION NUMBER OF THE HELP MODULES          *
*                                                                     *
*         VERSION OF DICTIONARY ONLINE DOCUMENTATION MODULES. PARAMETER
*         IS OPTIONAL, DEFAULT VALUE IS 1.                            *
*---------------------------------------------------------------------**
*                                                                     *
* (4)  : ==>   LOWCASE    LOWER CASE OPTION  (Y/N)                    *
*                                                                     *
*          INITIAL VALUE "LOWER CASE DATA ACCEPTED" OPTION ON THE DMLO*
*          SIGNON SCREEN.  VALUE IS CHANGEABLE DURING SESSION WITH    *
*          SET LOWCASE  (ON/OFF) COMMAND.                             *
*          ACCEPTABLE VALUES ARE 'Y' OR 'N' PARAMETER IS OPTIONAL,    *
*          DEFAULT VALUE 'N'.                                         *
*---------------------------------------------------------------------**
*                                                                     *
* (5)  : ==>   DPRTCL     DC PRINT CLASS                              *
*          INITIAL SETTING OF "PRINT CLASS" OPTION ON DMLO SIGNON     *
*          SCREEN.  VALUE IS CHANGEABLE DURING SESSION FROM OPTIONS   *
*          SCREEN.  PARAMETER IS OPTIONAL, DEFAULT VALUE '1'.         *
*---------------------------------------------------------------------**
*                                                                     *
* (6)  : ==>   TPRTCL     TSO PRINT CLASS                             *
*                                                                     *
*          INITIAL SETTING OF "PRINT CLASS" OPTION ON DMLO SIGNON     *
*          SCREEN.  VALUE IS CHANGEABLE DURING SESSION FROM OPTIONS   *
*          SCREEN.  PARAMETER IS OPTIONAL, DEFAULT VALUE 'A'.         *
*---------------------------------------------------------------------**
*                                                                     *
* (7)  : ==>   CPRTCL     CICS PRINT CLASS                            *
*                                                                     *
*          INITIAL SETTING OF "PRINT CLASS" OPTION ON DMLO SIGNON     *
*          SCREEN.  VALUE IS CHANGEABLE DURING SESSION FROM OPTIONS   *
*          SCREEN.  PARAMETER IS OPTIONAL, DEFAULT VALUE 'A'.         *
*                                                                     *
*          *** PRINTING IS NOT CURRENTLY AVAILABLE FROM CICS ***      *
*                                                                     *
*---------------------------------------------------------------------**
*                                                                     *
* (8)  : ==>   DISPLAY    DISPLAY FMT  (COBOL/VERTICAL)               *
*                                                                     *
*          INITIAL SETTING OF DISPLAY OPTION FOR DMLO SESSION.        *
*                                                                     *
*          VALUE IS CHANGEABLE DURING SESSION WITH  SET COBOL (ON/OFF)*
*          COMMAND.                                                   *
*                                                                     *
*          ACCEPTABLE VALUES ARE 'COBOL' OR 'VERTICAL'                *
*            WHERE COBOL    = LEVELED, INDENTED, COBOL-LIKE FORMAT    *
*                  VERTICAL = R4.6 AND EARLIER NON-LEVELED FORMAT     *
*                                                                     *
*          PARAMETER IS OPTIONAL, DEFAULT VALUE 'COBOL'.              *
*---------------------------------------------------------------------**
```

```
*                                                                     *
* (9)  : ==>   AUTOHEX    AUTOHEX   OPTION (ON/OFF)                   *
*                                                                     *
*          INITIAL SETTING OF AUTOHEX OPTION FOR DMLO SESSION.        *
*                                                                     *
*          VALUE IS CHANGEABLE DURING SESSION WITH                    *
*          SET AUTOHEX (ON/OFF) COMMAND.                              *
*                                                                     *
*          ACCEPTABLE VALUES ARE 'ON' OR 'OFF'                        *
*            WHERE  ON   ==> FIELDS CONTAINING INVALID DATA WILL BE   *
*                            AUTOMATICALLY DISPLAYED IN HEX FORMAT.   *
*                                                                     *
*          PARAMETER IS OPTIONAL, DEFAULT VALUE 'ON'.                 *
*---------------------------------------------------------------------**
*                                                                     *
* (10) : ==>   AUTOBND    AUTO-BIND OPTION (ON/OFF)                   *
*                                                                     *
*          INITIAL SETTING OF AUTOBIND OPTION FOR DMLO SESSION.       *
*                                                                     *
*          VALUE IS CHANGEABLE DURING SESSION WITH                    *
*          SET AUTOBIND (ON/OFF) COMMAND.                             *
*                                                                     *
*          ACCEPTABLE VALUES ARE 'ON' OR 'OFF'                        *
*            WHERE  ON   ==> RECORDS WILL BE AUTOMATICALLY BOUND      *
*                            AT THE FIRST REFERENCE IN DML COMMANDS   *
*                                                                     *
*          PARAMETER IS OPTIONAL, DEFAULT VALUE 'ON'.                 *
*---------------------------------------------------------------------**
*                                                                     *
* (11) : ==>   MAPIN      DATA/COMMAND INP (FAST/STEP)                *
*                                                                     *
*          INITIAL SETTING OF  MAPIN   OPTION FOR DMLO SESSION.       *
*                                                                     *
*          VALUE IS CHANGEABLE DURING SESSION                         *
*             WITH  SET MAPIN  (FAST/STEP) COMMAND                    *
*                                                                     *
*          ACCEPTABLE VALUES ARE 'FAST' OR 'STEP'                     *
*            WHERE FAST  ==> DATA UPDATES AND COMMAND/PFKEY INPUT     *
*                            WILL BE ACCEPTED IN THE SAME             *
*                            PSEUDO-CONVERSE.                         *
*                                                                     *
*          NOTE THAT RELEASES PRIOR TO R5.5 DMLO ONLY                 *
*               FUNCTIONED IN 'STEP' MODE.                            *
*                                                                     *
*          PARAMETER IS OPTIONAL, DEFAULT VALUE 'FAST'                *
*---------------------------------------------------------------------**
```

```
*                                                                     *
* (12) : ==>   CLIST      CLIST EXECUTION  (FAST/STEP)                *
*                                                                     *
*          INITIAL SETTING OF CLIST EXECUTION OPTION FOR SESSION.     *
*                                                                     *
*          VALUE IS CHANGEABLE DURING SESSION                         *
*             WITH  SET CLIST  (FAST/STEP) COMMAND                    *
*                                                                     *
*          ACCEPTABLE VALUES ARE 'FAST' OR 'STEP'                     *
*            WHERE FAST  ==> CLIST EXECUTION WILL BE IN FAST MODE.    *
*                                                                     *
*          PARAMETER IS OPTIONAL, DEFAULT VALUE 'FAST'                *
*---------------------------------------------------------------------**
*                                                                     *
* (13) : ==>   DSPCMND    COMMAND DISPLAY  (INPUT/USED)               *
*                                                                     *
*          INITIAL SETTING OF COMMAND DISPLAY OPTION FOR SESSION.     *
*                                                                     *
*          VALUE IS CHANGEABLE DURING SESSION                         *
*             WITH  SET CMND (INPUT/USED)  COMMAND                    *
*                                                                     *
*          ACCEPTABLE VALUES ARE 'USED' OR 'INPUT'                    *
*            WHERE USED  ==> COMMAND LINE ECHO WILL BE IN THE EXPANDED*
*                            FORMAT AS USED BY THE COMMAND PROCESSOR  *
*                  INPUT ==> COMMAND LINE ECHO WILL BE AS ENTERED     *
*                                                                     *
*          PARAMETER IS OPTIONAL, DEFAULT VALUE 'INPUT'               *
*---------------------------------------------------------------------**
*                                                                     *
* (14) : ==>   LRFSCRN    LRF SCREEN FMT   (NORM/MAX)                 *
*                                                                     *
*          INITIAL SETTING OF LRF SCREEN FORMAT OPTION.               *
*                                                                     *
*          VALUE IS CHANGEABLE DURING SESSION                         *
*             WITH      SHOW OPTIONS      COMMAND                     *
*                                                                     *
*          ACCEPTABLE VALUES ARE 'NORM' OR 'MAX'                      *
*            WHERE NORM  ==> SCREEN FORMAT FOR LRF SUBSCHEMAS INITIALLY
*                            WILL BE STANDARD 'EXPERT' FORMAT         *
*                                                                     *
*                  MAX   ==> SCREEN FORMAT FOR LRF SUBSCHEMAS INITIALLY
*                            WILL ALLOW FOR MAXIMUM COMMAND LENGTH    *
*                                                                     *
*          PARAMETER IS OPTIONAL, DEFAULT VALUE 'NORM'                *
*---------------------------------------------------------------------**
```

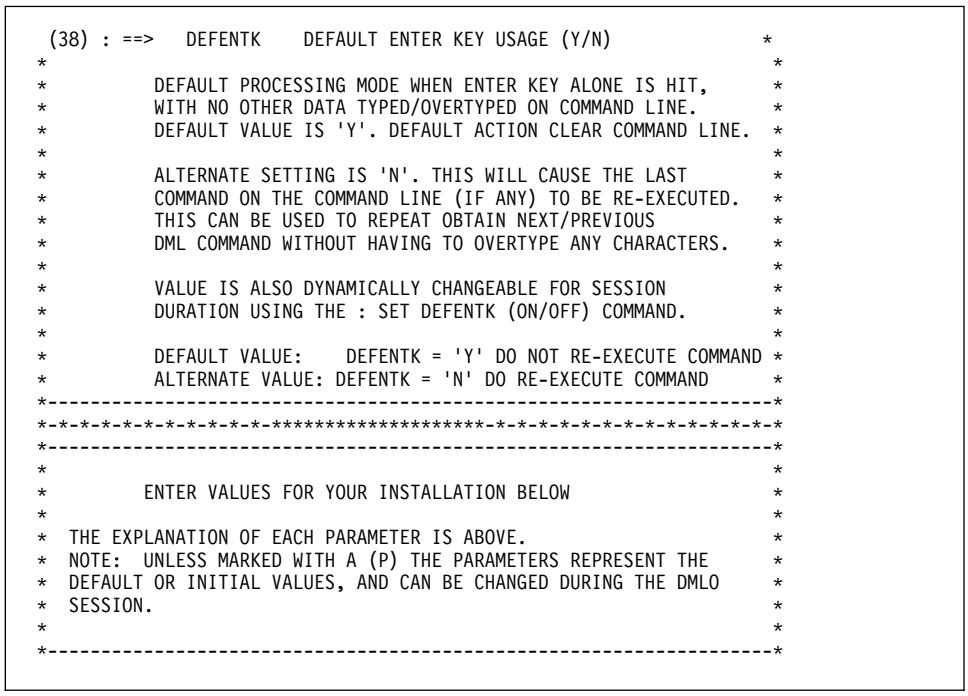
```
*                                                                     *
* (15) : ==>   MODE       SESSION MODE    (EXPERT/MENU)               *
*                                                                     *
*          INITIAL DEFAULT SETTING FOR MENU-MODE OPERATION.           *
*                                                                     *
*          VALUE IS CHANGEABLE DURING SESSION                         *
*             WITH  SET MENU (ON/OFF)      COMMAND                   *
*                                                                     *
*          ACCEPTABLE VALUES ARE 'NEMU' OR 'EXPERT'                   *
*            WHERE MENU   ==> DMLO WILL STARTUP IN MENU-MODE FORMAT   *
*                  EXPERT ==> DMLO WILL STARTUP IN EXPERT    FORMAT   *
*                                                                     *
*          PARAMETER IS OPTIONAL, DEFAULT VALUE 'EXPERT'              *
*---------------------------------------------------------------------**
*                                                                     *
* (16) : ==>   USERXIT    USER EXIT OPTION                            *
*                                                                     *
*          USER EXIT OPTION.                                          *
*          PARAMETER IS OPTIONAL, DEFAULT VALUE 'NO'                  *
*          ACCEPTABLE VALUES ARE :                                    *
*                                                                     *
*            YES           = USER EXIT MODULE IS AVAILABLE, WILL BE   *
*                            INVOKED FOR EACH DML VERB EXECUTED, AND  *
*                            OPTION IS NOT CHANGEABLE BY THE USER DURING
*                            THE DMLO SESSION                         *
*                                                                     *
*            NO            = USER EXIT WILL NOT BE INVOKED, AND OPTION*
*                            IS NOT CHANGEABLE BY THE USER DURING THE *
*                            DMLO SESSION                             *
*                                                                     *
*            (DYNAMIC,OFF) = USER EXIT MODULE IS AVAILABLE, BUT IS    *
*            (DYNAM,OFF)     NOT ACTIVE FOR THE SESSION UNTIL THE SET *
*                            EXIT ON COMMAND IS ISSUED. ALL DML VERBS *
*                            ARE ELIGIBLE FOR THE EXIT UNLESS         *
*                            SPECIFICALLY TURNED OFF DURING THE SESSION.
*                                                                     *
*                                                                     *
*            (DYNAMIC,ON)  = USER EXIT MODULE IS AVAILABLE, AND WILL  *
*            (DYNAM,ON)      BE ACTIVE FOR THE SESSION UNTIL THE SET  *
*                            EXIT OFF COMMAND IS ISSUED. ALL UNLESS   *
*                            SPECIFICALLY TURNED OFF DURING THE SESSION.
*                                                                     *
*            DYNAMIC       = EQUIVALENT TO (DYNAMIC,OFF)              *
*            DYNAM                                                    *
*---------------------------------------------------------------------**
*                                                                     *
* (17) : ==>   GLOBID     GLOBAL (SYS OWNED) PROFILE/CLIST OWNER      *
*                                                                     *
*          INTERNAL OWNER ID FOR GLOBAL (SYSTEM-OWNED)                *
*          PROFILES AND CLISTS                                        *
*                                                                     *
*          VALUE IS CHANGEABLE ONLY BY REASSEMBLY OF USDTPARM         *
*                                                                     *
*          PARAMETER OPTIONAL, DEFAULT VALUE 'DMLOSYS'                *
*                                                                     *
```

```
*-------------------------------------------------------------------**
*                                                                   *
* (18) : ==>   ADMIN      DMLO ADMINISTRATOR SIGNON (1)             *
* (19) : ==>   ADMIN2     DMLO ADMINISTRATOR SIGNON (2)             *
*                                                                   *
*          SIGNON USERIDS FOR WHICH DMLO WILL ALLOW RESTRICTED PROFILE*
*          AND CLIST MAINTENANCE FUNCTIONS                          *
*                                                                   *
*          VALUE IS CHANGEABLE ONLY BY REASSEMBLY OF USDTPARM       *
*                                                                   *
*          PARAMETER OPTIONAL, DEFAULT VALUES 'USERID01' 'USERID02' *
*                                                                   *
*-------------------------------------------------------------------**
*                                                                   *
* (20) : ==>   USERID     CHG USERID ?  (INPUT/PROT)                *
*                                                                   *
*          INDICATES WHETHER USERID FROM IDMS/DC SIGNON MAY BE BE   *
*          CHANGED AT DMLO SESSION SIGNON.                          *
*                                                                   *
*          VALUES ARE : INPUT   ==>  USERID/PASSWORD MAY BE ENTERED ON*
*                                    THE DMLO SIGNON SCREEN         *
*                       PROT    ==>  USERID/PASSWORD PROTECTED ON THE *
*                                    DMLO SIGNON SCREEN             *
*                                                                   *
*          PARAMETER IS OPTIONAL, DEFAULT VALUE  INPUT              *
*-------------------------------------------------------------------**
*                                                                   *
* (21) : ==>   NONDSPL    NONDISPLAY TRANSLATION                    *
*                                                                   *
*          INITIAL VALUE FOR TRANSLATION OF CHARACTERS WHICH ARE    *
*          CONSIDERED TO BE NON-DISPLAYABLE BASED ON CONTENTS OF TABLE*
*          DESCRIBED BY MEMBER USD@DSPC                             *
*                                                                   *
*          VALUE IS CHANGEABLE DURING SESSION                       *
*             WITH  SET NONDISPLAY X  COMMAND                       *
*                                                                   *
*          PARAMETER IS OPTIONAL, DEFAULT VALUE C'_'                *
*-------------------------------------------------------------------**
*                                                                   *
* (22) : ==>   DEFDICT    DEFAULT SIGNON DICTIONARY                 *
* (23) : ==>   DEFNODE    DEFAULT SIGNON DICT. NODE                 *
*                                                                   *
*         DEFAULT SIGNON SCREEN DICTIONARY NAME.                    *
*         DEFAULT SIGNON SCREEN DICTIONARY NODE.                    *
*                                                                   *
*         ANY VALUES SPECIFIED HERE WILL APPEAR EACH TIME THE DMLO  *
*         SIGNON SCREEN IS PRESENTED                                *
*-------------------------------------------------------------------**
```

```
*                                                                     *
* (24) : ==>   PRFDBNM    PROFILE SEGMENT (DB) NAME                   *
* (25) : ==>   PRFDBND    PROFILE SEGMENT (DB) NODE                   *
*                                                                     *
*         SEGMENT NAME AND NODE (DBNAME/DBNODE) FOR PROFILE/CLIST     *
*         SUBSCHEMA                                                   *
*                                                                     *
*         PARAMETERS SET BY INSTALLATION PROCESS TO MATCH DMCL CHANGES.
*---------------------------------------------------------------------**
*                                                                     *
* (26) : ==>   SBUFNM     DEFAULT SCRATCH REC NAME PREFIX             *
*                                                                     *
*         PREFIX FOR DEFAULT SCRATCH RECORD NAMES                     *
*                                                                     *
*         IF NO OTHER RECORD NAME SPECIFIED FOR SCRATCH I/O REQUESTS, *
*         DMLO WILL CREATE A RECORD/ELEMENT STRUCTURE WHOSE NAME IS   *
*         SSSSN WHERE SSSS IS SPECIFIED BY SBUFNM AND N IS 0-9.       *
*                                                                     *
* (27) : ==>   QBUFNM     DEFAULT QUEUE   REC NAME PREFIX             *
*                                                                     *
*         PREFIX FOR DEFAULT  QUEUE  RECORD NAMES                     *
*                                                                     *
*         IF NO OTHER RECORD NAME SPECIFIED FOR  QUEUE  I/O REQUESTS, *
*         DMLO WILL CREATE A RECORD/ELEMENT STRUCTURE WHOSE NAME IS   *
*         QQQQN WHERE QQQQ IS SPECIFIED BY QBUFNM AND N IS 0-9.       *
*                                                                     *
* (28) : ==>   SQBUFL     DEFAULT SCR/QUE REC MAX LEN                 *
*                                                                     *
*         DEFAULT SCRATCH/QUEUE BUFFER LENGTH                         *
*                                                                     *
*         THIS VALUE IS THE BUFFER LENGTH FOR ALL RECORDS WHICH DMLO  *
*         ALLOCATES USING THE DEFAULT SCRATCH AND QUEUE RECORD NAME   *
*         PREFIXES.                                                   *
*                                                                     *
*---------------------------------------------------------------------**
*                                                                     *
* (29) : ==>   ATTNKEY    ATTENTION/INTERRUPT                         *
*                                                                     *
*         INITIAL VALUE OF "INTERRUPT" KEY WHICH APPEARS ON THE DMLO  *
*         SIGNON SCREEN. NOTE IT IS CHANGEABLE AT THAT TIME.  ACCEPTABLE
*         VALUES ARE PA1-PA3, OR PF1-PF24.                            *
*                                                                     *
*         PARAMETER IS OPTIONAL, DEFAULT VALUE IS 'PA1'.              *
*---------------------------------------------------------------------**
*                                                                     *
* (30-37)   THESE REPRESENT INITIAL VALUES OF PF KEY SETTINGS.        *
*           ALLOWED FORMATS ARE :                                     *
*                  ....KEY=(PFX,PFY)                                  *
*                  ....KEY=(PFX)                                      *
*                  ....KEY=PFX                                        *
*                                                                     *
*           ALL KEYS ARE CHANGEABLE DURING THE SESSION EXCEPT         *
*                  SNONKEY == SIGNON SCREEN HELP                      *
*                  PROFKEY == SIGNON PROFILE LIST                     *
```

```
*-------------------------------------------------------------------**
*                                                                   *
* (30) : ==>   SNONKEY    HELP (SIGNON)                             *
*                                                                   *
*         INITIAL VALUE OF PRIMARY AND ALTERNATE PF KEYS USED TO INVOKE
*         ONLINE DOCUMENTATION FOR SIGNON SCREEN                    *
*                                                                   *
*         KEYS CHANGEABLE ONLY AT INSTALLATION                      *
*                                                                   *
*         ACCEPTABLE VALUES ARE PF1-PF24.                           *
*                                                                   *
*         DEFAULT VALUES : 'PF2'    AND  'PF14'                     *
*-------------------------------------------------------------------**
*                                                                   *
* (31) : ==>   PROFKEY    SIGNON PROFILE LIST                       *
*                                                                   *
*         INITIAL VALUE OF PRIMARY AND ALTERNATE PF KEYS USED TO REQUEST
*         A PROFILE LIST FROM SIGNON SCREEN                         *
*                                                                   *
*         KEYS CHANGEABLE ONLY AT INSTALLATION                      *
*                                                                   *
*         ACCEPTABLE VALUES ARE PF1-PF24.                           *
*                                                                   *
*         DEFAULT VALUES : 'PF4'    AND  'PF16'                     *
*------------------------------------------------------------------- *
*                                                                   *
* (32) : ==>   HELPKEY    HELP (DMLO)                               *
*                                                                   *
*          INITIAL VALUE OF PRIMARY AND ALTERNATE PF KEYS USED TO   *
*          INVOKE DMLO ONLINE DOCUMENTATION (HELP) DISPLAYS.        *
*                                                                   *
*          KEYS CHANGEABLE DURING SESSION.                          *
*                                                                   *
*          ACCEPTABLE VALUES ARE PF1-PF24.                          *
*                                                                   *
*          DEFAULT VALUES : 'PF1'    AND  'PF13'                    *
*------------------------------------------------------------------- *
*                                                                   *
* (33) : ==>   SHOWKEY    SHOW PFKEYS                              *
*                                                                  *
*          INITIAL VALUE OF PRIMARY AND ALTERNATE PF KEYS USED TO   *
*          REQUEST DISPLAY/UPDATE OF ALL PF KEYS.                   *
*                                                                   *
*          KEYS CHANGEABLE DURING THE SESSION                       *
*                                                                   *
*          ACCEPTABLE VALUES ARE PF1-PF24.                          *
*                                                                   *
*          DEFAULT VALUES : 'PF2'    AND  'PF14'                    *
*-------------------------------------------------------------------*
```

```
*                                                                     *
* (34) : ==>   PENDKEY    END / GOBACK FUNCTION                       *
*                                                                     *
*          INITIAL VALUE OF PRIMARY AND ALTERNATE PF KEYS USED TO     *
*          REQUEST END/GOBACK FROM SECONDARY DMLO DISPLAYS            *
*                                                                     *
*          KEYS CHANGEABLE DURING THE SESSION                         *
*                                                                     *
*          ACCEPTABLE VALUES ARE PF1-PF24.                            *
*                                                                     *
*          DEFAULT VALUES : 'PF3'    AND  'PF15'                      *
*---------------------------------------------------------------------*
*                                                                     *
* (35) : ==>   DISPKEY    DISPLAY &D                                  *
*                                                                     *
*          INITIAL VALUE OF PRIMARY AND ALTERNATE PF KEYS USED TO     *
*          REQUEST REDISPLAY FUNCTION.                                *
*                                                                     *
*          KEYS CHANGEABLE DURING THE SESSION                         *
*                                                                     *
*          ACCEPTABLE VALUES ARE PF1-PF24.                            *
*                                                                     *
*          DEFAULT VALUES : 'PF4'    AND  'PF16'                      *
*---------------------------------------------------------------------*
*                                                                     *
* (36) : ==>   PGUPKEY    SCROLL UP                                   *
*                                                                     *
*          INITIAL VALUE OF PRIMARY AND ALTERNATE PF KEYS USED TO     *
*          PAGE/SCROLL DISPLAY UP (TOWARD THE FIRST LINE)             *
*                                                                     *
*          KEYS CHANGEABLE DURING THE SESSION                         *
*                                                                     *
*          ACCEPTABLE VALUES ARE PF1-PF24.                            *
*                                                                     *
*          DEFAULT VALUES : 'PF7'    AND  'PF19'                      *
*---------------------------------------------------------------------*
*                                                                     *
* (37) : ==>   PGDNKEY    SCROLL DOWN                                 *
*                                                                     *
*          INITIAL VALUE OF PRIMARY AND ALTERNATE PF KEYS USED TO     *
*          PAGE/SCROLL DISPLAY DOWN  (TOWARD THE LAST LINE)           *
*                                                                     *
*          KEYS CHANGEABLE DURING THE SESSION                         *
*                                                                     *
*          ACCEPTABLE VALUES ARE PF1-PF24.                            *
*                                                                     *
*          DEFAULT VALUES : 'PF8'    AND  'PF20'                      *
*---------------------------------------------------------------------*
*                                                                     *
```


(38) : ==> DEFENTK DEFAULT ENTER KEY USAGE (Y/N)
DEFAULT PROCESSING MODE WHEN ENTER KEY ALONE IS HIT,
WITH NO OTHER DATA TYPED/OVERTYPED ON COMMAND LINE.
DEFAULT VALUE IS 'Y'. DEFAULT ACTION CLEAR COMMAND LINE.
ALTERNATE SETTING IS 'N'. THIS WILL CAUSE THE LAST
COMMAND ON THE COMMAND LINE (IF ANY) TO BE RE-EXECUTED.
THIS CAN BE USED TO REPEAT OBTAIN NEXT/PREVIOUS
DML COMMAND WITHOUT HAVING TO OVERTYPE ANY CHARACTERS.
VALUE IS ALSO DYNAMICALLY CHANGEABLE FOR SESSION
DURATION USING THE : SET DEFENTK (ON/OFF) COMMAND.
DEFAULT VALUE: DEFENTK = 'Y' DO NOT RE-EXECUTE COMMAND
ALTERNATE VALUE: DEFENTK = 'N' DO RE-EXECUTE COMMAND
ENTER VALUES FOR YOUR INSTALLATION BELOW
THE EXPLANATION OF EACH PARAMETER IS ABOVE.
NOTE: UNLESS MARKED WITH A (P) THE PARAMETERS REPRESENT THE
DEFAULT OR INITIAL VALUES, AND CAN BE CHANGED DURING THE DMLO
SESSION.

```
     USDCPARM HLPDICT=TOOLDICT,      1 DICTNAME FOR ONLINE DCMNTN (P) X
              HLPNODE=,              2 DICTNODE FOR ONLINE DCMNTN (P) X
              HLPVERS=1,             3 VERSION  FOR ONLINE DCMNTN (P) X
              LOWCASE=N,             4 LOWER CASE OPTION  (Y/N)       X
              DPRTCL=1,              5 DC PRINT CLASS                 X
              TPRTCL=A,              6 TSO PRINT CLASS                X
              CPRTCL=A,              7 CICS PRINT CLASS               X
              DISPLAY=COBOL,         8 DISPLAY FMT  (COBOL/VERTICAL)  X
              AUTOHEX=ON,            9 AUTOHEX   OPTION (ON/OFF)      X
              AUTOBND=ON,           10 AUTO-BIND OPTION (ON/OFF)      X
              MAPIN=FAST,           11 DATA/COMMAND INP (FAST/STEP)   X
              CLIST=FAST,           12 CLIST EXECUTION  (FAST/STEP)   X
              DSPCMND=INPUT,        13 COMMAND DISPLAY  (INPUT/USED)  X
              LRFSCRN=NORM,         14 LRF SCREEN FMT   (NORM/MAX)    X
              MODE=EXPERT,          15 SESSION MODE    (EXPERT/MENU)  X
              USERXIT=(DYNAM,OFF),  16 USER EXIT OPTION               X
              GLOBID=DMLOSYS,       17 SYS. PROF/CLIST OWER ID        X
              ADMIN=USERID01,       18 DMLO ADMINISTRATOR SIGNON      X
              ADMIN2=USERID02,      19 DMLO ADMINISTRATOR SIGNON      X
              USERID=INPUT,         20 CHG USERID ?  (INPUT/PROT)     X
              NONDSPL=C'_',         21 NONDISPLAY TRANSLATION         X
              DEFDICT=,             22 DEFAULT SIGNON DICTIONARY      X
              DEFNODE=,             23 DEFAULT SIGNON DICT. NODE      X
              PRFDBNM=DMLO,         24 PROFILE SEGMENT (DB) NAME      X
              PRFDBND=,             25 PROFILE SEGMENT (DB) NODE      X
              SBUFNM=SBUF,          26 DEFAULT SCR REC NAME PFX       X
              QBUFNM=QBUF,          27 DEFAULT QUE REC NAME PFX       X
              SQBUFL=4096,          28 DEFAULT S/Q REC MAX LEN        X
              ATTNKEY=PA1,          29 ATTENTION/INTERRUPT            X
              SNONKEY=(PF2,PF14),   30 SIGNON HELP           (P)      X
              PROFKEY=(PF4,PF16),   31 PROFILE LIST          (P)      X
              HELPKEY=(PF1,PF13),   32 SESSION HELP                   X
              SHOWKEY=(PF2,PF14),   33 SHOW PFKEYS                    X
              PENDKEY=(PF3,PF15),   34 END                            X
              DISPKEY=(PF4,PF16),   35 REDISPLAY                      X
              PGUPKEY=(PF7,PF19),   36 SCROLL UP                      X
              PGDNKEY=(PF8,PF20),   37 SCROLL DOWN                    X
              DEFENTK=Y             38 DEFUALT USE OF ENTER KEY
*-------------------------------------------------------------------*
*                                                                   *
*     1) ALTER ANY/ALL OF THE PRECEEDING USDCPARM PARAMETERS, AS    *
*        NEEDED.  (A DETAILED EXPLANATION IS PRESENTED ABOVE.)      *
*                                                                   *
*     2) ALTER ANY/ALL OF THE FOLLOWING SOURCE MODULES (AS NEEDED)  *
*        FOR YOUR INSTALLATION BEFORE ASSEMBLY OF USDPARM1.  (A     *
*        DETAILED EXPLANATION IS PRESENTED WITHIN EACH MODULE       *
*                                                                   *
*-------------------------------------------------------------------*
        COPY  USD@MOPS            MENU-MODE DML OP CODES
        COPY  USD@MTXT            MENU-MODE DESCRIPTIVE TEXT
        COPY  USD@MSTL            MENU-MODE STATIC AREA DESCRIPTION
        COPY  USD@SSEX            SUBSCHEMA EXCLUSION LIST
        COPY  USD@DSPC            DISPLAYABLE CHARACTERS
        COPY  USD@KYWD            STANDARD ABBREVIATIONS
        END
```

G.8 CA-IDMS/Enforcer Runtime Parameters

```
*------------------------------------------------------------------------
*CA-IDMS/ENFORCER RUNTIME PARAMETERS
*------------------------------------------------------------------------
*            MODIFY PRODUCT TUNING PARAMETERS
*            ________________________________
*ESXTPARM ── THIS MEMBER IS USED TO SPECIFY THE RUNTIME VALUES TO
*            BE USED AS INPUT TO THE INSTALLATION TAILORING MACRO,
*            ESXCPARM, WHICH IS DYNAMICALLY LOADED BY ONLINE MENU
*            ESXAMEN, AND THE ACTIVE AND PASSIVE ENFORCEMENT MODULES.
*
*            RUNTIME VARIABLES
*            _________________
*            ENFTSK=(1-8 CHAR)    TASK USED TO INVOKE THE ENFORCER.
*
*            HLPDICT=(1-8 CHAR)   ALTERNATE DICTIONARY USED
*                                 FOR GSIHELP.
*
*            HLPNODE=(1-8 CHAR)   ALTERNATE NODE USED FOR GSIHELP.
*
*            HLPVERS=(1-9999 NUM) VERSION NUMBER OF HELP MODULES.
*
*            LOKMODE=(D/B/M)      IDD DEADLOCK PROCESSING DIRECTIVE
*                                 WHERE:
*                                 D = DEADLOCK--ALLOW FULL ENFORCER
*                                     DIAGNOSTICS.  THIS MODE WILL
*                                     CAUSE DEADLOCKS AGSINST CON-
*                                     CURRENT UPDATE OF THE SAME ENTITY
*                                     TYPE IN THE SAME DICTIONARY.
*                                 B = BATCH MODE--ONLY ALLOWS IDD-
*                                     FORMAT ERROR MESSAGES BUT
*                                     PRECLUDES DEADLOCK ERRORS.
*                                 M = IDDM ONLY--ALLOWS FULL
*                                     ENFORCER DIAGNOSTICS FOR IDDM
*                                     TRANSACTIONS. ALL OTHER
*                                     PROCESSING IS IDENTICAL TO BATCH
*                                     MODE.
*
*USPS: NEW PARMS TO INDICATE WHICH DELIMITERS ARE VALID FOR ELEMENT
*      DESIGNATION FOR BRACKET MODE TEMPLATES.
*            DSPACE=(Y/N)          SPACE DELIMITED WORDS ALLOWED.
*                                  Y = YES (ALLOWED)
*                                  N = NO  (ALLOWED)
*            DDASH=(Y/N)           DASH (-) DELIMITED WORDS ALLOWED.
*                                  Y = YES (ALLOWED)
*                                  N = NO  (ALLOWED)
*            DULINE=(Y/N)          ULINE(_) DELIMITED WORDS ALLOWED.
*                                  Y = YES (ALLOWED)
*                                  N = NO  (ALLOWED)
*------------------------------------------------------------------------
*  DEFAULT VALUES AS SUPPLIED WITH INSTALLATION:
*       ESXTPARM ENFTSK='ENFORCER',
*                HLPDICT='        ',
*                HLPNODE='        ',
*                HLPVERS=1,
*                LOKMODE=D,
*USPS: DELIMITER VALUES FOR BRACKET TEMPLATING.
*                DSPACE=Y,
*                DDASH=Y,
*                DULINE=Y
*------------------------------------------------------------------------
```

G.9 CA-IDMS/Master Key Runtime Parameters

```
*--------------------------------------------------------------------
* CA-IDMS/MASTERKEY RUNTIME PARAMETERS
*--------------------------------------------------------------------
*
*SSKCPARM THIS IS THE INSTALLATION TAILORING MACRO USED BY THE
*MASTERKEY SYSTEM TO PROVIDE RUNTIME VALUES.
*OPERANDS:
*                 HLPDICT='1-8 CHAR'   ALTERNATE DICTIONARY USED FOR
*                                      GSIHELP.
*                 HLPNODE='1-8 CHAR'   ALTERNATE NODE USED FOR GSIHELP.
*
*                 HLPVERS=INTEGER      VERSION NUMBER OF HELP MODULES.
*
*                 CLTDICT='1-8 CHAR'   DICTNAME FOR TRANSIENT CLISTS.
*
*                 CLTNODE='1-8 CHAR'   DICTNODE FOR TRANSIENT CLISTS.
*
*ASSEMBLED VALUES AT INSTALLATION:
*       SSKCPARM HLPDICT='        ',
*                HLPNODE='        ',
*                HLPVERS=1,
*                CLTDICT='        ',
*                CLTNODE='        '
*
*--------------------------------------------------------------------
```

G.10 CA-IDMS/Online Log Display Runtime Parameters

```
 *--------------------------------------------------------------------
 * CA-IDMS/ONLINE LOG DISPLAY RUNTIME PARAMETERS
 *--------------------------------------------------------------------
 *
         PRINT OFF
         COPY  USKCPARM
         PRINT ON
USKTPARM CSECT                        CONTROL TABLE FOR LOGD
 *-------------------------------------------------------------------*
 *USKCPARM THIS IS THE INSTALLATION TAILORING MACRO USED BY THE
 *LOGD SYSTEM TO PROVIDE RUN-TIME VALUES.
 *OPERANDS:
 *                 LOGDTSK='1-8 CHAR'   TASK USED TO INVOKE LOGD.
 *                 HLPDICT='1-8 CHAR'   ALTERNATE DICTIONARY USED FOR
 *                                      GSIHELP.
 *                 HLPNODE='1-8 CHAR'   ALTERNATE NODE USED FOR GSIHELP.
 *
 *                 HLPVERS=INTEGER      VERSION NUMBER OF HELP MODULES.
 *
 *ASSEMBLED VALUES AT INSTALLATION:
 *       USKCPARM LOGDTSK='LOGD    ',
 *                HLPDICT='        ',                              *
 *                HLPNODE='        ',                              *
 *                HLPVERS=1                                        *
 *                                                                 *
 *-------------------------------------------------------------------*
 ********************************************************************
 *                                                                  *
 *******         MODIFY THE FOLLOWING STATEMENTS IF NEEDED       *******
 *                                                                  *
 ********************************************************************
         USKCPARM LOGDTSK=LOGD,                                     X
               HLPDICT=TOOLDICT,                                    X
               HLPNODE=,                                            X
               HLPVERS=,
         END
 *--------------------------------------------------------------------
```

G.11 CA-IDMS/SASO Runtime Parameters

```
*--------------------------------------------------------------------
* CA-IDMS/SASO RUNTIME PARAMETERS
*--------------------------------------------------------------------
*
*            MODIFY PRODUCT TUNING PARAMETERS
*            ________________________________
*ESSTPARM —— THIS MEMBER IS USED TO SPECIFY THE RUNTIME VALUES TO
*            BE USED AS INPUT TO THE INSTALLATION TAILORING MACRO,
*            ESSCPARM, WHICH IS DYNAMICALLY LOADED BY SASO PRODUCT
*            INSTALLATION UTILITY FUNCTION TO INITIALLY ESTABLISH
*            ONLINE SYSTEM DEFAULTS AND BY ONLINE MENU ESSAMENU.
*
*            RUNTIME VARIABLES
*            _________________
*            SASOTSK=(1-8 CHAR)   TASK USED TO INVOKE SASO.
*
*            HLPDICT=(1-8 CHAR)   ALTERNATE DICTIONARY USED FOR
*                                 GSIHELP.
*
*            HLPNODE=(1-8 CHAR)   ALTERNATE NODE USED FOR GSIHELP.
*
*            HLPVERS=(1-9999 NUM) VERSION NUMBER OF HELP MODULES.
*
*            DEFDOC=(1-8 CHAR)    DEFAULT DOCUMENT DATABASE
*                                 NAME TABLE ENTRY (DBNAME).
*
*            JCL1=('1-79 CHAR')   DEFAULT JCL JOB CARD LINES
*                                 FOR INITIAL USER PROFILES.
*
*            JCL2=('1-79 CHAR')   DEFAULT JCL JOB CARD LINES
*                                 FOR INITIAL USER PROFILES.
*
*            JCL3=('1-79 CHAR')   DEFAULT JCL JOB CARD LINES
*                                 FOR INITIAL USER PROFILES.
*
*            NOTE: JCL VALUES MUST BE ENCLOSED IN SINGLE QUOTES.
*                  DEFAULT VALUES DO NOT REQUIRE QUOTES.
*
*  DEFAULT VALUES AS SUPPLIED WITH INSTALLATION:
*      ESSCPARM SASOTSK='SASO    ',
*               HLPDICT='        ',
*               HLPNODE='        ',
*               HLPVERS=1,
*               DEFDOC='SPG     ',
*               JCL1=' ',
*               JCL2=' ',
*               JCL3=' '
*
*  NOTE: NULL SPECIFICATION OF JCL LINES CAUSES SASO TO USE THE
*         PREDEFINED DEFAULTS RELATED TO THE RUNTIME OPERATING
*         SYSTEM IN WHICH THE PRODUCT IS INSTALLED.
*--------------------------------------------------------------------
```

G.12 General Sort Runtime Parameters

```
*--------------------------------------------------------------------
* CA-IDMS/DC-SORT RUNTIME PARAMETERS
*--------------------------------------------------------------------
*
*TPSCPARM THIS IS THE INSTALLATION TAILORING MACRO USED BY THE
*TP/SORT FACILITY TO PROVIDE RUNTIME STORAGE LIMITS AND ALGORITHMS
*NECESSARY TO DISTRIBUTE THE SORT WORK RECORDS INTO THE SORTED
*OUTPUT.
*OPERANDS:
*                 MAIN=(0 THRU N)      AMOUNT OF MAIN STORAGE
*                                      USED FOR INTERNAL SORT BUFFERS.
*                                      ZERO IS AN ALL-SCRATCH SORT.
*                 AUX=(0 THRU N)       MAXIMUM AMOUNT OF SCRATCH STORAGE
*                                      USED BY CA-IDMS/DC SORT.
*                 MINRBUF=(0-N)        MINIMUM DESIRABLE NUMBER OF RECORDS
*                                      IN A SORT BUFFER.
*                 LIMLOCK=(Y OR N)     (Y)ES OR (N)O PREVENT INDIVIDUAL
*                                      PROGRAMS FROM EXCEEDING INSTALLATION
*                                      LIMITS
*                 EXIT=(PA1..PF24)     ADS PREPROCESSOR EXIT
*                                      KEY DEFAULT IS PA2.
*
*EACH OF THE SUPPLIED VALUES MUST BE AN INTEGER CONSTANT.
*THE NUMBER REPRESENTS THE ACTUAL NUMBER OF BYTES.
*    10000        = 10,000 BYTES
*    100000       = 100,000 BYTES
*
*THIS PLACES A RESPONSIBILITY UPON THE INSTALLER TO KEEP IT WITHIN
*REASON.
*
*ASSEMBLER VALUES AT INSTALLATION:
*       TPSCPARM MAIN=10000,AUX=10000,MINRBUF=100,LIMLOCK=N,
*             EXITKEY=PA2
*
*EXAMPLE
*
*MAIN=10000
*AUX=10000
*MINRBUF=20
*record-length=100
*
*The sort buffer used by CA-IDMS/DC SORT will be 2012 bytes:
*
*      20 * 100 = 2000
*      2000 is a multiple of 2000
*      20000 + 12 = 2012
*
*CA-IDMS/DC SORT CAN STORE FOUR SORT BUFFERS (80 RECORDS) IN A MAIN STORAGE
*OF 10,000 BYTES AND FOUR SORT BUFFERS (80 RECORDS) IN SCRATCH (AUXILIARY)
*STORAGE OF 10,000 BYTES.
*
```

Appendix H. CA-IDMS/DMLO Implementations

H.1 CA-IDMS/DMLO Security and Access H-4
H.1.1 CA-IDMS/DMLO Security H-4
H.1.2 CA-IDMS/DMLO Access Restrictions H-5
H.1.2.1 Restricting Usage Mode Access Globally H-5
H.1.2.2 Restricting Usage Mode Access by User H-5
H.1.2.3 Central CA-IDMS Security H-6
H.2 Implementing CA-IDMS/DMLO in Multiple CV's Under CICS H-7

This appendix discusses the security and access restrictions available for dictionaries containing subschemas that are accessed using DMLO. It also discusses the steps required to install CA-IDMS/DMLO for multiple CV's under CICS.

H.1 CA-IDMS/DMLO Security and Access

This appendix describes security and access restrictions that can be applied to dictionaries containing subschemas to be accessed using CA-IDMS/DML Online.

H.1.1 CA-IDMS/DMLO Security

CA-IDMS/DMLO provides security checking on three levels.

- **Level 1** security indicates that a security check is not needed. Any user who signs on to CA-IDMS/DMLO and specifies a valid subschema for the requested dictionary is permitted to access the database. Level 1 is the default security level.
- **Level 2** security indicates that CA-IDMS/DMLO verifies that the user and password combination specified during CA-IDMS/DMLO sign-on exist in the requested dictionary. If they do, the user can access any valid subschema in that dictionary.
- **Level 3** security indicates that CA-IDMS/DMLO not only validates the user and password, but also verifies that the user has authorization to access the requested subschema. The user must be registered for access to the requested subschema in the requested dictionary.

Use the following syntax to register for access to a given subschema:

```
(ADD/MOD) USER userid PASSWORD pswd
INCLUDE ACCESS TO SUBSCHEMA subname OF SCHEMA schname V vers-nbr.
```

For both Level 2 and Level 3 security, special consideration is given to situations where the user ID used to sign on to the CA-IDMS/DMLO session is the same as the user ID used to sign on to the CA-IDMS/DC system. In this case, the password is not checked even though the user must still be defined to the requested dictionary. Non-validation of the password conforms to the processing done by the dictionary task.

To implement security for CA-IDMS/DMLO, you must register program DBMSDMLO with a version number of 1, 2 or 3. The version number corresponds to the desired security level. Use the following syntax to add this program:

```
ADD PROGRAM NAME IS DBMSDMLO VERSION IS n.
```

You must register DBMSDMLO in each dictionary for which security beyond the default is required.

H.1.2 CA-IDMS/DMLO Access Restrictions

CA-IDMS/DMLO has six possible usage modes:

- SR — Shared Retrieval
- SU — Shared Update
- PR — Protected Retrieval
- PU — Protected Update
- ER — Exclusive Retrieval
- EU — Exclusive Update

You can restrict the READY modes available both globally (all users in a given dictionary) and by user. Any such restrictions are applied each time a user request is made to ready an area.

H.1.2.1 Restricting Usage Mode Access Globally

To restrict access to specific usage modes for all users for all subschemas in a given dictionary, use the PROGRAM DESCRIPTION clause of the ADD PROGRAM statement.

Example:

```
ADD/MOD PROGRAM DBMSDMLO VERSION IS 1
    PROGRAM DESCRIPTION IS 'SR,PR,ER'.
```

With this example, Level 1 security is established, but only retrieval modes are allowed for any subschema within the dictionary with this registration.

Note: When you specify more than one usage mode:

- Abbreviations must be separated by commas
- Cannot contain any imbedded blanks
- The string must be enclosed in single quotation marks

H.1.2.2 Restricting Usage Mode Access by User

To restrict usage mode access by user within a given dictionary, you must have specified Level 2 or Level 3 security for that dictionary.

For each user with particular restrictions, you must specify the allowable usage modes with the USER DESCRIPTION clause.

Example: For a specified user cannot access any subschemas in the given dictionary with other than “shared” access modes.

```
ADD/MOD USER userid PASSWORD pswd USER DESCRIPTION IS 'SR,SU' .
```

H.1.2.3 Central CA-IDMS Security

Remember the centralized CA-IDMS security facility supersedes any validation by CA-IDMS/DMLO. That is, if access to a dictionary or database is prohibited by the central security facility, you **cannot** use CA-IDMS/DMLO to bypass or override that level of security.

H.2 Implementing CA-IDMS/DMLO in Multiple CV's Under CICS

This appendix describes how to install CA-IDMS/DMLO on multiple CV's under CICS. Perform the following steps:

1. Create additional copies of the load module/relocatable USDTPIF5, naming the copied version with a unique suffix (for example, USDTPIFA).
2. Add entries to the PPT for each of the programs created in Step 1. Target source library member USDPPT can be modified for this purpose.
3. Add a unique transaction code to the PCT for each version of USDTPIF*a*[1]. The first three characters of the transaction codes **must** be unique to CA-IDMS/DMLO. The fourth character of each transaction code must be unique to that version of CA-IDMS/DMLO. Target source library member USDPCT can be modified for this purpose.
4. Link the copied version of the USDTPIF*a*[1] to include the appropriate copy of IDMSCINT to interface with the desired IDMS CV.

   CWIF0SLD member USDLNKCS can be used as a template for the link step.

   [1] Where *a* is a unique suffix described in Step 1.

Index

Special Characters

#UCFCICS 7-9
#UCFDEND 7-9
#UCFUFT 7-9

A

ACCEPT 6-10, 8-14, 10-8
Add-On variables 5-11
Allocate 6-4, 6-16, 8-4, 10-4
APARs 12-4, D-10, D-17, D-22
APPLDICT 6-18
APPLID 2-7, 7-6
APPLY 6-9, 8-13, 10-7, 12-4, D-8
APPLY CHECK 12-4
ASF 6-18
authorized libraries 7-12

B

back-end 7-7, 7-8
Backup 5-10, 6-12, 6-23

C

CA-C 2-8
CA-Culprit 5-11
CA-IDMS Release 10.2 7-19
CA-IDMS/ADS ALIVE 8-6
CA-IDMS/ADS Trace 9-6
CA-IDMS/DATABASE EXTRACTOR 8-6, 9-3
CA-IDMS/DC SORT 8-7, 9-6
CA-IDMS/DICTIONARY MIGRATOR 8-6, 9-6
CA-IDMS/Dictionary Migrator Assistant 9-3
CA-IDMS/DICTIONARY MODULE EDITOR 8-6
CA-IDMS/DICTIONARY QUERY FACILITY 8-6
CA-IDMS/DML ONLINE 8-6, 8-7
CA-IDMS/DMLO 8-7
  access restrictions H-5
  security H-4
CA-IDMS/ENFORCER 8-6
CA-IDMS/MASTERKEY 8-6
CA-IDMS/ONLINE LOG DISPLAY 8-6
CA-IDMS/SASO 8-6
CA-IDMS\Test Database Builder 9-3
CAIIPDS 6-5, 8-5, 10-5
  return codes 6-5, 8-5, 10-5
CAIRIM 5-12, 6-14, 7-15
CAISAG 4-1, 4-4, 4-9
  dynamic PDE allocation 5-9
  parameters C-23
  static PDE allocation 5-9
CAS9 3-7, 7-15
CICS 2-7, 5-4, 7-7
CICS access 7-7
CICS PPT 7-7, 7-10
CICS systems
  debugging E-9
  IDMSCINT module E-12
  IDMSINTC module E-11
CICS, CWA 5-11
CICSPPT 7-7
COBOL 2-8
comment 4-7
Commonweather 6-19
Cover letter 3-5
CPU 3-7
  requirements 2-4
Customization 8-7
CV number 5-7, 7-8
CWA 5-11, 7-7, 7-8

D

database 6-19
Database Files 6-16
demonstration 6-19
Dictionary Files 6-16
DICTIONARY MIGRATOR ASSISTANT 9-3
disk contention 5-5
DMCL 5-4
DMLO 8-7
Download 4-4
dynamic PDE allocation 5-9

E

elements 6-8, 10-6

F

FREESTG 5-7
front-end 7-7
front-end table 7-9, 7-10

G

GENASM 6-11, 8-15, 12-4
GENERAL SORT 8-6
GIM23901E D-18
GIM39701W 6-8
GIM41101E 13-19
GJF0INIT 6-14
Global 5-4
Global DMCL 5-4, 6-17
GSISVCX 8-11

I

I/O, runtime 5-11
IDMSCINT 7-7
IDMSCINT module E-12
IDMSINTC 6-22, 7-8
IDMSOPTI 6-6
IDMSPPT 7-7
IEBCOPY B-4
IEBGENER B-4
IEW2303S D-18
IEW2456W D-18
IEW2648E D-18
indirect files 6-5, 8-5, 10-5, D-7
installation 3-1
  introduction 1-1, 2-1

J

JCL 4-4
JES3 6-4
Job 09 6-13, 8-16
Job 1 6-4, 8-4, 10-4
Job 10 6-14
Job 11 6-16
Job 13 6-18
Job 14 6-19
Job 15 6-20
Job 16 6-21
Job 17 6-22
Job 18 6-23
Job 19 6-24
Job 2 6-5, 8-5, 10-5
Job 3 6-6, 8-6, 10-6
Job 4 6-8, 8-11, 10-7
Job 5 6-9, 8-12, 10-8
Job 6 6-10, 8-13, 10-9
Job 7 6-11, 8-14
Job 8 6-12, 8-15
journals 5-5

K

Key Certificate 3-6
keys 3-6

L

library
  authorized 7-12
  requirements 2-8
  sample JCL 12-8
LMP keys 3-6

M

macro 6-6
macro parms 6-6
maintenance
  materials 12-7
  sample JCL library 12-8
  steps
    APPLY CA-IDMS Maintenance 12-10
    APPLY CHECK CA-IDMS Maintenance 12-10
    Customize the SMP/E Procedure 12-9
    Load Maintenance Sample JCL Library 12-8
    overview 12-8
    Re-APPLY Applicable SYSMODs 12-12
    RECEIVE CA-IDMS Maintenance 12-10
    RESTORE applicable SYSMODs 12-10
    Save all Materials and Output 12-13
    Special processing maintenance 12-13
maintenance tape 12-7

N

non-swappable 7-12

O

optional APARs 7-17

P

page size 5-11
PASSWORD 4-6
passwords 5-4
PDE 5-9
performance 5-7, 6-15

PML 3-5
product code 3-7
Product Maintenance Letter 3-5
products 5-4
program pool 5-11
PTF 12-7

R

RECEIVE 6-8, 8-12, 10-6, 12-4
REFRESH 6-15
REJECT 12-5
RESTORE 12-5
RESTORE BYPASS(ID) 12-5
RHDCFSTB 7-8
RHDCOPTF 7-17
RHDCPARM 5-7, 7-11
RHDCSRTT 6-17
RHDCUCFT 7-6
Runtime options G-1

S

S806
  abend 6-15
SAMPJCL 3-4, 4-4, 6-24
sample JCL B-4
SETUPBK 4-7
SFnn
  abend 6-17
SMP/E 2-8, 4-3, 5-10, 6-4, 6-11, 6-12, 8-4, 8-12, 10-4, 12-4
  backups 5-6
  CA-DISPATCH 5-6
  commands
    ACCEPT 8-14, 10-8
    APPLY 8-13, 10-7
  CSI 5-6
  environment 5-6
  release, prior 5-6
SQL 6-20
SQL Demonstration database 6-20
startup 5-11, 6-13
STARTUP JCL 6-9
Startup Module 5-8
static PDE allocation 5-9
SVC 2-6, 5-7, 6-17, 7-15, 7-16, 8-11, 9-3
  installing 6-14
  SVC installation 6-14
SYSDIRL 6-17
System 90 6-17, 6-21
System 99 6-17
SYSTEM dictionary 6-17
system verification 7-12
systems programmer 5-11, 7-15

T

tape 2-5, 4-4
tape contents
  backup files B-4
  sample JCL PDS B-4
test fixes 12-4
TPNAME 7-8
TSO 7-6

U

UCFCICS 6-22
Unicenter TNG Framework for OS/390 2-6, 5-12
Upgrade install types 5-5, 5-13
Uppercase terminal support
userid 5-8
USERMOD 6-6, 6-13, 7-18, 13-3
  building 13-5
  installing 13-13
USERMODs 3-9

V

value 4-7
VARBLIST 4-7
  customizing parameters 3-4
variable 4-7
verification demo 7-12
VSAM 5-8
VTAM 2-7, 5-10, 7-6
VTAM ID 5-10
VTAMLIN 2-7, 7-6

W

WTOEXIT 5-8, 6-13